

DTIC FILE COPY

(1)

AD-A224 761

Prototype Message Dissemination System  
and  
Document Routing System  
for an  
Army Organization

DTIC  
ELECTE  
AUG 02 1990  
S & D D

by  
Pamela O. Howard

**DISTRIBUTION STATEMENT A**

Approved for public release  
Distribution Unlimited

A Report Submitted in Partial Fulfillment  
of the Requirement for the Degree  
of Master of Science,  
Management Information Systems  
at the University of Arizona  
May 1990

96 08 26 012

Copy the following files from the disks into the same directory as the runtime version of EXSYS Professional (version 1.1.4). At the DOS prompt, type 'GO.' Then, follow the instructions in the User's guide.

These files must be present to run the Message Dissemination System (MDS) and the Document Routing System (DRS).

GO.BAT            A batch file that calls the necessary sequence of programs.

REPORT.EXE       A report formatting program developed by Mike Morrison.  
Syntax: REPORT filename.RPT "Header Info"  
See batch file for use here.

NEWREP.HLP       Note: REPORT.EXE must have a file called filename.HLP  
in order to work. This is a popup help window.  
This .hlp file is included with the program for  
that purpose.

TEMP.DBF         These files must be present for the query program  
REPORT.DBF       to work.  
QUERY.DBF

MAILISC.TXT      These files are expert system files that work in  
MAILISC.RUL      conjunction with EXSYS Professional. They were  
MAILISC.CFG      built specifically to support USAISC.  
MAILISC.CMD  
MAILISC.OUT

MAIL.PAS         This is the source code and the executable code for the  
MAIL.EXE         Pascal programs that are called to work with the MDS and  
COMBINE.PAS       DRS. They are called by the batch file.  
COMBINE.EXE

QUERY.PRG        This is the source code and the executable code for the  
QUERY.EXE        dBase III plus program which builds queries and provides  
a list of EMAIL recipients.

EXSYSP.OVL       These are the EXSYS Professional (version 1.1.4) files  
EXSYSP.HLP       required for the program to operate. Since EXSYS  
EXSYSP.COM       Professional is a copy protected program, any future  
EXSYSP.EXE       users must purchase a copy of the program. It cannot  
be included with this research project.

Return.dat       Do not be alarmed to find these files after running  
Report.rpt       the program. They are built by the various executable  
Report.txt       programs for transferring or displaying data.  
Newrep.rpt  
Hdr.dat  
Pass.dat

ISCTOP.DBF       These are the .dbf files for the dBase III plus  
ISCDIR.DBF       database. They were created exclusively for use  
ISCDIV.DBF       with this program.  
ISCBR.DBF  
ISCEMP.DBF  
ISCPROJ.DBF  
ISCBRPRO.DBF  
ISCEMPPR.DBF

DIREMP.DBF       These .dbf files were created through joining  
DVEMP.DBF       one or more of the above files.  
BREMP.DBF  
BRPROCH.DBF

~~FILE 3~~

# Contents

|          |   |           |
|----------|---|-----------|
| <b>1</b> | <b>Introduction</b>                             | <b>4</b>  |
| <b>2</b> | <b>Architecture</b>                             | <b>5</b>  |
| 2.1      | KMS Description . . . . .                       | 5         |
| 2.2      | KMS Architecture . . . . .                      | 6         |
| <b>3</b> | <b>Conceptual System Description</b>            | <b>7</b>  |
| 3.1      | Message Dissemination . . . . .                 | 7         |
| 3.2      | Document Routing . . . . .                      | 8         |
| 3.3      | Database/Knowledge Base Coupling . . . . .      | 9         |
| <b>4</b> | <b>Knowledge Acquisition</b>                    | <b>10</b> |
| 4.1      | Organizational Knowledge . . . . .              | 10        |
| 4.2      | Organizational Communication Policies . . . . . | 10        |
| 4.2.1    | Scenario One . . . . .                          | 11        |
| 4.2.2    | Scenario Two . . . . .                          | 11        |
| 4.2.3    | Scenario Three . . . . .                        | 12        |
| 4.2.4    | Scenario Four . . . . .                         | 13        |
| 4.2.5    | Scenarios Five and Six . . . . .                | 13        |
| 4.2.6    | Scenario Seven . . . . .                        | 14        |
| 4.2.7    | Scenario Eight . . . . .                        | 14        |
| 4.2.8    | Scenario Nine . . . . .                         | 15        |
| 4.2.9    | Scenario Ten . . . . .                          | 15        |
| 4.2.10   | Scenario Eleven . . . . .                       | 16        |
| 4.2.11   | Scenarios Twelve and Thirteen . . . . .         | 16        |
| 4.2.12   | Scenarios Fourteen and Fifteen . . . . .        | 17        |
| <b>5</b> | <b>Design and Implementation</b>                | <b>17</b> |

|       |  |    |
|-------|--|----|
| 5.1   | Database Design . . . . .                            | 18 |
| 5.2   | Expert System Design . . . . .                       | 19 |
| 5.2.1 | Knowledge Management Module (KMM) . . . . .          | 20 |
| 5.2.2 | Corporate Knowledge Base (CKB) . . . . .             | 20 |
| 5.3   | User Interface Module (UIM) . . . . .                | 20 |
| 5.4   | Corporate Database Interface Module (CDIM) . . . . . | 21 |
| 5.5   | Process Example . . . . .                            | 22 |
| 6     | Future Directions                                    | 24 |
| 7     | Conclusion   | 28 |
| 7.1   | Advantages of an MDS . . . . .                       | 28 |
| 7.2   | Advantages of a DRS . . . . .                        | 28 |
| 8     | Appendix A   | 30 |
| 8.1   | Database Schema . . . . .                            | 30 |
| 8.2   | Rules Listing . . . . .                              | 30 |
| 8.3   | Go.bat . . . . .                                     | 30 |
| 8.4   | Mailisc.cmd . . . . .                                | 30 |
| 8.5   | Mailisc.out . . . . .                                | 30 |
| 8.6   | Mailisc.cfg . . . . .                                | 30 |
| 8.7   | Mail.pas . . . . .                                   | 31 |
| 8.8   | Combine.pas . . . . .                                | 31 |
| 8.9   | Newrep.hlp . . . . .                                 | 31 |
| 8.10  | Query.prg . . . . .                                  | 31 |
| 8.11  | Read.me . . . . .                                    | 31 |
| 9     | Appendix B - Figures                                 | 32 |
| 10    | Appendix C - User's Guide                            | 33 |
| 10.1  | Introduction . . . . .                               | 33 |

|                                    |    |
|------------------------------------|----|
| 10.2 System Requirements . . . . . | 33 |
| 10.3 Operation . . . . .           | 33 |
| 10.4 Sample Runs . . . . .         | 36 |
| 10.4.1 Sample MDS Run . . . . .    | 36 |
| 10.4.2 Sample DRS Run . . . . .    | 38 |

|                    |                                     |
|--------------------|-------------------------------------|
| Accession For      |                                     |
| NTIS CRA&I         | <input checked="" type="checkbox"/> |
| DTIC TAB           | <input type="checkbox"/>            |
| Unannounced        | <input type="checkbox"/>            |
| Justification      |                                     |
| By <i>per call</i> |                                     |
| Distribution/      |                                     |
| Availability Codes |                                     |
| Dist               | Avail and/or Special                |
| <i>A-1</i>         |                                     |

STATEMENT "A" per Capt Pamela Howard  
Class 90-8, Ft. Leavenworth, KS 66027-5021

TELECON

8/1/90

VG



# 1 Introduction

Electronic mail systems (EMS) are used by many different organizations, including the U.S. Army. An EMS has many advantages. It allows the action officer or the manager to be more productive because the action officer no longer has to play "telephone tag" in order to communicate with another individual. The mail is sent electronically, and the other individual replies (hopefully) whenever they receive the communication. An EMS also enhances productivity because the action officer is not required to make introductory and closing "small talk." The electronic mail (~~EMAIL~~) can be concise and to the point. Additionally, the action officer does not have to put the electronic mail into a special format, as is often required by other communication means, such as the standard Army letter or the standard Army message format.)

Unfortunately, there are two distinct disadvantages of using an EMS. First, the EMS is presently a "dumb" facility; it provides no automated mail dissemination or document routing. The action officer is required to know the individual and the individual's EMAIL identifier in order to send that individual EMAIL. Often, an EMAIL facility provides no assistance in this area. After an action officer is established within an organization, this may not be a problem; however, new action officers may have difficulty getting to know their counterparts (the individuals with whom they should be communicating), especially in large, diverse, and distributed organizations. Second, the EMAIL is presently an informal and more or less random means of communication; it provides no assistance with the routing of those formal documents or documents that require chain-of-command treatment.

In order to enhance support for management activities, Motiwalla et al. proposed the Knowledge-based Mail System (KMS) [7] which addressed the two issues discussed in the previous paragraph. This Master's project uses the conceptual design provided by Motiwalla et al. to develop a KMS which addresses the problems of mail dissemination and document routing for the United States Army Information Systems Command (USAISC). This paper describes the design and implementation of a prototype KMS system for USAISC. First, a brief description of the KMS architecture will be provided in the next section. Then the database and expert system designs will be discussed. The database/knowledgebase coupling mechanism will be examined next. Finally, future directions and conclusions will be presented.

This project is based on the investigation described in [5]. The KMS is a portion of the Integrated Office Information Systems (IOIS) Project at the Management Information Systems Department at the University of Arizona, Tucson, Arizona.

## 2 Architecture

### 2.1 KMS Description

The KMS was designed by Motiwalla et al. to enhance the traditional organizational EMS by incorporating knowledge regarding the organization and the individual into the system. Table 1 shows the proposed functions of a KMS [7].

| KMS-Functions             |
|---------------------------|
| Asynchronous GDSS (AGDSS) |
| Environmental Scanning    |
| Activity Planning         |
| Project Coordination      |
| Meeting Scheduling        |
| Alerting                  |
| Task Management           |
| Document Routing          |
| Message Dissemination     |
| Message Management        |

Table 1: The Functions of KMS

Each of the features of a KMS is briefly described below:

1. **Asynchronous Group Decision Support Systems (AGDSS).** This feature of the KMS would provide GDSS type services to a group in an asynchronous fashion. Given a topic, the KMS could use the organizational knowledge to develop an agenda, identify and notify the participants, provide beginning and ending times for the asynchronous session, etc. [7].
2. **Environmental Scanning.** This facility would automatically review electronic information for the users of an organization and based on organizational knowledge and user preferences, it could disseminate specific articles to the appropriate users [7].
3. **Activity Planning.** The KMS could manage routine activity planning for the manager, such as weekly meetings, leaving the non-routine activities for the manager to handle [7].

4. **Project Coordination.** A KMS could manage portions of the project coordination mission, such as reminding the user of deadlines and negotiation activities [7].
5. **Meeting Scheduling.** The KMS could perform meeting scheduling based on monitoring the calendars of involved personnel and placing priority on accommodating the highest ranking individuals [7].
6. **Alerting.** This feature would remind the user via an audio or a visual signal of upcoming meetings, appointments, etc [7].
7. **Task Management.** This feature of the KMS could manage the assignment of tasks to workers and monitor their progress for the manager [7].
8. **Document Routing.** This facility could use knowledge of the organizational structure to route documents for review and approval through the chain of command [7].
9. **Message Dissemination.** This feature of the KMS would use organizational knowledge about the structure of the organization and the organizational communication policies to develop recipient lists for the user based on the subject of the message [7].
10. **Message Management.** Based on the user's personal preferences, the KMS could sort and prioritize the incoming messages which would enable the user to read the most important messages first [7].

For more information and discussion on the features of the KMS, the reader is referred to [3, 4, 7, 8].

## 2.2 KMS Architecture

The KMS architecture developed by Motiwalla et al. is shown in Figure 1 of Appendix B. Those components enclosed by the circle were implemented for the message dissemination system (MDS) and the document routing system (DRS) developed for this project. Each of these architectural components is described below:

1. **User Interface Module (UIM).** This component accepts input from and delivers output to the user [7].
2. **Knowledge Management Module (KMM).** This is the inference engine portion of the expert system. It accepts information from the UIM and inferences that with information in the corporate database to develop the information required to build a query [7].



3. **Corporate Knowledge Base (CKB).** This is the portion of the expert system where the rules regarding organizational communication policies are stored [7].
4. **Corporate Database (CDB).** Data concerning the organizational structure and activities, such as projects, are stored and maintained in the CDB [7].
5. **Corporate Database Interface Module (CDIM).** This module will provide the interface between the CDB and KMM [7].

### **3 Conceptual System Description**

This project concentrates on two areas of the KMS, message dissemination and document routing, identified and described above, as applied to the United States Army Information Systems Command (USAISC) described in [5].

#### **3.1 Message Dissemination**

The Message Dissemination System (MDS) provides automated message dissemination to the user. It should overcome the first disadvantage listed in the background discussion, above. In other words, the MDS should relieve the action officer of the burden of knowing the individuals and their EMAIL identifiers who need to receive his/her communication. The MDS uses structured attributes, such as project names, job titles or job categories, and organizational activities obtained from the message header and inferences them with the CKB to develop the attributes for the query into the CDB. Using that information, the CDB then generates a distribution list for the user.

For this project, the MDS was implemented using the EXSYS Professional expert system and dBase III plus on a PC. The design information for the rules in the expert system and the facts in the database were developed in part from the information contained in [5], from personal experience working at USAISC, and also from interviews with an action officer from USAISC. The EMS interface was not implemented since this is a PC based prototype, and there was no available EMS for a PC network. The implementation strategies for the major components of the MDS and DRS as described above follow:

1. **Knowledge Management Module (KMM).** This component is provided by the EXSYS Professional expert system.
2. **Corporate Knowledge Base (CKB).** This component is implemented using rules in the EXSYS Professional expert system.

3. **Corporate Data Base (CDB).** This component is implemented using dBase III plus. The Structured Object Method (SOM) (previously known as the structured entity method (SEM)) is the design methodology for the database.
4. **Corporate Data Base Interface Module (CDIM).** This component is implemented using both a dBase III plus program and a Pascal interface routine.

The MDS interfaces with the user and the existing EMS with the following components:

1. **User Interface Module (UIM).** For the purpose of this project, a Pascal routine models a typical EMS user interface. This same Pascal routine parses (to a very limited extent) the information that the user provides into input that the EXSYS Professional expert system can interpret. The rest of the MDS and DRS are hidden from the user in the form of an embedded system. Finally, at the conclusion of the run, a second Pascal program adds time limits (when required) and header information to the list of recipients. The resulting list is then displayed to the user by a C program.
2. **EMS Interface Module (EIM).** This feature was not implemented because this project was implemented on a stand-alone PC.

Conceptually, the user provides a subject and a recipient from a subset of keywords which the Pascal program is capable of parsing. The subset of keywords is provided in the user's guide in Appendix C. This information is then provided to the KMM and the CKB; whereupon, the appropriate rules are fired. The MDS then accesses the CDB to identify the actual recipients of the message based on the user-provided responses. Once the MDS determines the recipient list, header information and time limits (when required) are added to the list. Then, it is displayed in an output screen for the user's approval. The software for the screen display was developed by Mike Morrison using the C programming language. This screen display tool is available for use by projects which fall under the IOIS umbrella. Ideally, after user approval, the list is forwarded to the EIM for dissemination over the existing EMS. However, this is a prototype MDS which does not include an EIM; therefore, the approved recipient list will be stored in a file rather than disseminated.

### 3.2 Document Routing

The Document Routing System (DRS) provides automated document routing to the user. It should overcome the second disadvantage listed in the introductory discussion in section one. The DRS should relieve the action officer of some of the physical work involved

in staffing a document for approval. Staffing a document involves acquiring the correct signatures for approval/disapproval and monitoring the progress of the document. The DRS would provide (to the associated EMS) the document and the *route* of the document through the organizational hierarchy to all levels necessary to gain final approval. The DRS will also use structured attributes, such as document title and staff level of approval, obtained from the message header. This information would be inferenced with the organizational knowledge base and database to generate the route for the document. The major components for the DRS would be the same as those described under the MDS above; however, the rules for the DRS in the knowledge base differ from those of the MDS.

Conceptually, the DRS operates in a fashion similar to the MDS described above. However, the list of EMAIL recipients generated for document routing is an ordered list starting with the lowest level chief and secretary, and proceeding through the intermediate levels to the highest level chief and secretary required for that particular document. A time limit for viewing the document is placed on each chief in the hierarchy.

### 3.3 Database/Knowledge Base Coupling

The database/knowledge base coupling provides the interface between the user application, the database management system (DBMS), the database, and the knowledge base management system and the knowledge base. Figure 2 in Appendix B depicts this interface.

There are several advantages to be gained in coupling a database and a knowledge base in stead of trying to store and maintain all of the data in the knowledge base. These benefits are described below:

1. **Insures Data Integrity.** A database and a DBMS are designed to maintain data integrity (consistency); an expert system is not designed for the same kind of data manipulation. By using a coupling mechanism, the DBMS is responsible for data storage and manipulation, and the knowledge base is only responsible for rule storage and manipulation.
2. **Reduces Redundancy.** Again, by separating the DBMS functions from the expert system functions, reduction of redundancy (a typical DBMS function) is performed naturally. Reducing redundancy in an expert system would be a difficult task as frequently rules refer to the same data.
3. **Reduces Maintenance Cost.** Another typical advantage of the DBMS is the reduction of maintenance costs due to the orderly storage and manipulation of data. An expert system usually does not depend on a particular order, so maintenance costs, especially if dealing with large quantities of data, would be expensive.

4. **Requires No/Minimal Transition Cost.** The coupling can be built for existing databases which keeps transition costs to a minimum.
5. **Facilitates Complex Query Navigation.** Since the expert system is not also attempting to manage large amounts of data, query navigation proceeds unimpeded.

## **4 Knowledge Acquisition**

Two kinds of knowledge were required for this project:

1. **Facts.** This is the organizational knowledge which associates personnel with job categories, organizational activities, and organizational structure. This information had to be acquired and arranged appropriately in the database. Facts are maintained in the CDB.
2. **Rules.** This pertains to the communication policies within USAISC which had to be acquired. Communication policies concern the protocols for disseminating messages within an organization. These policies, explicit and implicit, may be implemented with rules (in an expert system).

### **4.1 Organizational Knowledge**

Organizational knowledge consists of the facts regarding personnel, job classification categories, and organizational activities. This information was acquired via personal experience working in USAISC and interviews with another USAISC action officer. USAISC was analyzed via the use of the Structured Object Model (SOM). An in-depth discussion of the SOM methodology may be found in [1, 2]. The facts are maintained in tables in dBase III plus. For a listing of the relations, see Appendix A. The database design process for this project is described in section 5.1.

### **4.2 Organizational Communication Policies**

Organizational Communication Policies (rules) were acquired in terms of scenarios. Fifteen case studies were developed based on information acquired from USAISC. The first ten case studies apply to the MDS. The final five scenarios apply to the DRS. Each case study was analyzed to determine what kind of knowledge would be needed to implement that scenario in the MDS or DRS. For the MDS case studies, each represents a situation in which a user needs to propagate a message to a subset of the personnel in USAISC regarding a particular topic. For the DRS case studies, each represents a

situation in which a user at the lowest level of the chain of command needs to propagate a document upward through each intermediate level. These scenarios represent typical communication policies and messages within USAISC. Each scenario is described in the following sections. This list is by no means all inclusive. As the understanding of the communication policies improves, more rules may be included. A listing of the rules is included at Appendix A.

The SOM diagram is useful for understanding inference patterns. An inference pattern identifies which items need to be known in order for other items to be discovered. Inference patterns are very helpful for developing the rules. For example, (see the SOM diagram, figure 3, in Appendix B) the aspects of the "Project" entity are branch and employee. Imagine a circle around these two aspects and the entity project; that is the inference pattern. If the user knows the name of the project, by using the inference pattern, the user can determine the employees and branches (offices) involved with that project. Understanding and using the inferencing patterns of the SOM diagram clarifies the rule-building process.

#### 4.2.1 Scenario One

The first scenario concerns an announcement for a funding meeting for primary project officers. This is a typical situation for a funding meeting to be called by the resource managers of a large military organization like USAISC. The primary project officers manage the money for their projects; however, due to normal turnover, it may be difficult for the resource managers to maintain a list of primary project officers. For this situation, the MDS must identify the recipients of the message and their branch chiefs from the information provided by the user. The policy implemented in this situation is:

- Whenever primary project officers are notified of a meeting, their branch chiefs should be notified simultaneously. This enables branch chiefs to better manage their personnel.

Using the subject, "FUNDING MEETING," and the recipient, "PRIMARY PROJECT OFFICERS," the MDS will then access the database to determine which personnel are designated primary project officers and to select their appropriate branch chiefs as well. The computer IDs (for EMS purposes) and projects of these personnel and their branch chiefs are then placed into a file containing the names of the potential recipients. Once the user has validated the list of recipients, it would be distributed via the EMS.

#### 4.2.2 Scenario Two

The second scenario concerns the dissemination of information regarding training courses. The recipients for this information are all division and branch chiefs of one directorate.

Normally, this type of information is distributed (by the personnel managers) down the chain of command; however, it is a suspense (limited amount of time to respond) action. The directorate chief (or his or her secretary) does not have the time to list every branch and division chief name when transmitting this type of information. In this situation, broadcasting the information to branch and division chiefs speeds delivery of the information to the levels where it is needed. The MDS identifies all branch and division chiefs in that directorate in the database. The policy implemented in this situation is:

- Whenever all branch and division chiefs of a directorate are to be notified of training courses, the MDS will select all branch and division chiefs in that directorate from the database for notification.

Using the subject, "TRAINING COURSES," and the recipient, "DIVISION & BRANCH CHIEFS/ASPL," the MDS will then access the database to determine which personnel are designated branch and division chiefs in the directorate "ASPL". The computer IDs and titles of these personnel are then placed into a file containing the names of the potential recipients, etc. It is important to understand that most information should traverse the chain of command one step at a time, i.e. no broadcast. However, the announcement of training courses is administrative in nature, and it does not require formal chain of command communication. Any information which falls into this category could be treated in the same fashion.

#### 4.2.3 Scenario Three

The third scenario concerns the dissemination of a meeting announcement. The recipients for this information are all division chiefs of one directorate. Again, the directorate chief (or his/her secretary) does not have the time to list every division chief name when transmitting this type of information. In this situation, broadcasting the information to division chiefs speeds delivery of the information to the levels where it is needed. The MDS identifies all division chiefs in that directorate in the database. The policy implemented in this situation is:

- Whenever all division chiefs of a directorate are to be notified of a meeting, the MDS will select all division chiefs and division secretaries in that directorate from the database for notification. Typically, secretaries maintain the calendars for the division chiefs, and they can immediately annotate the calendar with the date and time of the meeting.

Using the subject, "MEETING," and the recipient, "DIVISION CHIEFS/ASPL," the MDS will then access the database to determine which personnel are designated division

chiefs and secretaries in the directorate "ASPL". The computer IDs and titles of these personnel are then placed into a file containing the names of the potential recipients, etc. Any information which is normally managed by the secretaries could be treated in the same fashion.

#### 4.2.4 Scenario Four

The fourth scenario concerns the dissemination of some other information to division chiefs. The recipients for this information are all division chiefs of one directorate. The purpose of this scenario is to demonstrate that some information would only be sent to the division chiefs without automatically informing their secretaries. For example, this could be privileged (although not classified) information. Specific topics could easily be designated upon user request. Again, the directorate chief may not be interested in taking the time to list every division chief name when transmitting this type of information. In this situation, broadcasting the information to division chiefs speeds delivery of the information without sacrificing privacy. The MDS identifies all division chiefs in that directorate in the database. The policy implemented in this situation is:

- Whenever all division chiefs of a directorate are to be notified of subjects which are not for general information, the MDS will select all division chiefs in that directorate from the database for notification.

Using a subject (any subject will do), "TSQ-XXX," and the recipient, "ALL DIVISION CHIEFS/ASRM," the MDS will then access the database to determine which personnel are designated division chiefs in the directorate "ASRM". The computer IDs and titles of these personnel are then placed into a file containing the names of the potential recipients, etc.

#### 4.2.5 Scenarios Five and Six

The fifth and sixth scenarios concern the dissemination of a meeting announcement or a suspense action to all branch chiefs within a division. The recipients for this information are all branch chiefs and their respective secretaries within a division. Since the secretaries normally maintain the calendars for the branch chiefs, they can immediately annotate the calendar with the date and time of the meeting. The branch secretaries normally monitor suspense actions for the branch chief; by receiving the suspense information at the same time as the branch chief, the branch chief need only provide the name of the action officer (assigned to the suspense action) to the secretary. The MDS identifies all branch chiefs and secretaries in that division in the database. The policy implemented in this situation is:

- Whenever all branch chiefs of a division are to be notified of a meeting or a suspense action, the MDS will select all branch chiefs and their secretaries in that division from the database for notification.

Using the subject, "WEEKLY MEETING" or "SUSPENSE ACTION," and the recipient, "BRANCH CHIEFS/ASPL-A," the MDS will then access the database to determine which personnel are designated branch chiefs or secretaries in the division "ASPL-A". The computer IDs and titles of these personnel are then placed into a file containing the names of the potential recipients, etc.

#### 4.2.6 Scenario Seven

The seventh scenario concerns the dissemination of private information to directorate chiefs from a higher level. The recipients for this information are all of the directorate chiefs in USAISC. The purpose of this scenario is to demonstrate another method for disseminating private information, i.e. information that would not automatically be sent to directorate secretaries at the same time. Again, this could be privileged (although not classified) information. Specific topics could easily be designated upon user request. A time saving is achieved in that the sender is not required to list every directorate chief's name when transmitting. The MDS identifies all directorate chiefs in the database. The policy implemented in this situation is:

- Whenever all directorate chiefs are to be notified of subjects designated as private, the MDS will select all directorate chiefs in the database for notification. Secretaries are not to be notified.

Using a subject (any subject will do), "RIF," and the recipient, "DIRECTORATE CHIEFS/PRIV," the MDS will then access the database to determine which personnel are designated directorate chiefs in the database. The computer IDs and directorates of these personnel are then placed into a file containing the list of the potential recipients.

#### 4.2.7 Scenario Eight

The eighth scenario concerns the dissemination of other information to directorate chiefs from a higher level. The recipients for this information are all of the directorate chiefs and their secretaries in USAISC. This scenario demonstrates that information not designated as private could automatically be sent to directorate secretaries at the same time. A time savings is achieved in that the sender is not required to list the names of all the directorate chief's and their secretaries when transmitting. The MDS identifies all directorate chiefs and secretaries in the database. The policy implemented in this situation is:



- Whenever all directorate chiefs are to be notified of subjects not designated as private, the MDS will select all directorate chiefs and their secretaries in the database for notification.

Using a subject (any subject will do), "ANYSUB," and the recipient, "DIRECTORATE CHIEFS," the MDS will then access the database to determine which personnel are designated directorate chiefs or directorate secretaries in the database. The computer IDs and directorates of these personnel are then placed into a file containing the list of potential recipients.

#### 4.2.8 Scenario Nine

The ninth scenario concerns the dissemination of information regarding a specific project to project officers. The sender, for example, could be a resource manager concerned about the funding of a project or an operations managers concerned about the status of a project. The recipients for this information are all of the project officers (also known as action officers) designated as primary or alternate for this project. A time savings is achieved in that the sender is not required to list every project officer's name when transmitting. The policy implemented in this situation is:

- Whenever all project officers are to be notified of information regarding their projects, the MDS will select all project officers (primary and alternate) in the database for notification.

Using a subject (any subject will do), "MEETING," and the recipient, "PROJECT OFFICERS/AWIS," the MDS will then access the database to determine which personnel are designated project officers (primary or alternate) for this project in the database. Each project officer's computer ID and project are placed in the list of potential recipients along with the associated branch chief computer IDs and titles.

#### 4.2.9 Scenario Ten

The tenth scenario concerns the dissemination of information to all employees of USAISC. The sender, for example, could be a security manager announcing the annual security briefing which all employees are required to attend. The recipients for this information are all employees. All personnel in the database will be notified of the briefing. This kind of a scenario could easily be modified to notify only all military personnel or only all civilian personnel in the database of some information. The policy implemented in this situation is:

- Whenever all employees are to be notified of information, the MDS will select all personnel in the database for notification.

Using a subject (any subject will do), "SECURITY BRIEFING," and the recipient, "ALL EMPLOYEES," the MDS will then access the database and notify all personnel of this topic.

#### 4.2.10 Scenario Eleven

The eleventh scenario concerns document routing from the branch level to the division level. The sender would typically be an action officer. The recipients for this information are in order: the action officer's branch chief and secretary and the division chief and secretary. A time savings is achieved in that the sender is not required to list the chiefs and secretaries in his/her chain of command. The policy implemented in this situation is:

- Whenever document routing is required from branch to division level, the DRS will produce an ordered list of recipients consisting of the branch chief and secretary and then the division chief and secretary for notification.

Using a subject, "FYI," and the qualified recipient, "DOC RTNG - DIV/ASPL-AA," the DRS will then access the database to determine the branch chief and secretary for the "ASPL-AA" branch and then the respective division chief and secretary. The computer IDs and titles of these personnel are then placed into a file containing the names of the potential recipients. Since this is document routing, a time limit is also placed next to the names of the recipients. For the purpose of this project, the recipients are given one day to review the document. This is not a realistic time, but it demonstrates the concept.

#### 4.2.11 Scenarios Twelve and Thirteen

These two scenarios are concerned with document routing from the branch level to the directorate level. Again, the sender would typically be an action officer. The recipients for this information are in order: the action officer's branch chief and secretary, then the respective division chief and secretary, then the respective directorate chief and secretary. Again, a time savings is achieved in that the sender is not required to list the chiefs and secretaries in his/her chain of command. The policy implemented in this situation is:

- Whenever document routing is required from branch to directorate level, the DRS will produce an ordered list of recipients consisting of the branch chief and secretary, then the division chief and secretary, and then the directorate chief and secretary for notification.

Using one of the following subjects: "WEEKLY ACTIVITY REPORT" or "DOCUMENT APPROVAL," and the qualified recipient, "DOC RTNG - DIR/ASPL-AA," the DRS will then access the database to determine the branch chief and secretary for the "ASPL-AA" branch, then the respective division chief and secretary, and then the respective directorate chief and secretary. The computer IDs and titles of these personnel are then placed into a file containing the names of the potential recipients. Again, a time limit is placed next to the names of the recipients.

#### 4.2.12 Scenarios Fourteen and Fifteen

These two scenarios are concerned with document routing from the branch level to the commander (general officer) level. Again, the sender would typically be an action officer. The recipients for this information are in order: the action officer's branch chief and secretary, the respective division chief and secretary, the respective directorate chief and secretary, and then the Commander of USAISC and secretary. Again, a time savings is achieved in that the sender is not required to list the chiefs and secretaries in his/her chain of command. The policy implemented in this situation is:

- Whenever document routing is required from branch to commander level, the DRS will produce an ordered list of recipients consisting of the branch chief and secretary, then the division chief and secretary, then the directorate chief and secretary, and then the Commander of USAISC and secretary for notification.

Using one of the following subjects: "INFO PAPER" or "BLUE BULLET," and the qualified recipient, "DOC RTNG - GENERAL/ASPL-AA," the MDS will then access the database to determine the branch chief and secretary for the "ASPL-AA" branch, then the respective division chief and secretary, then the respective directorate chief and secretary, and then the Commander of USAISC and secretary. The computer IDs and titles of these personnel are then placed into a file containing the names of the potential recipients. Again, an additional time limit is also placed next to the names of the recipients. A "blue bullet" is a request for information from the general regarding a specific subject. The turn-around time varies, but it is usually within 48 hours. Since the turn-around time varies, it may be difficult to place exact time limits on this kind of information.

## 5 Design and Implementation

This project is a PC-based prototype Message Dissemination System (MDS) and Document Routing System (DRS), which is implemented using the EXSYS Professional expert system and the dBase III plus database management system (DBMS). There were three significant operating requirements for this design.

1. The PC environment was mandated by the IOIS project. The requirements of this environment include DOS 2.0 or higher and an 80286 machine running at 10 MHZ, minimum. A color monitor and a printer are optional.
2. Due to the PC environment, the system cannot use greater than 640 kilobytes of main memory. Ideally, this is one portion of a larger system. The total system will eventually also have to accommodate the memory requirements for the resident EMS system and networking software as well, so the MDS and the DRS must use significantly less than 640 kilobytes of main memory. Because of the size of the expert system, the database and the DBMS, a batch file is used to move the different software in and out of main memory.
3. Efficiency (in terms of speed of execution) is also a major consideration. Since the user will be waiting for the MDS/DRS to respond, the MDS/DRS should consume only a minimum amount of time. This is a difficult requirement due to the complexity and time involved for interfacing the expert system and the database.

The rest of this section describes the design of the MDS/DRS in detail.

## 5.1 Database Design

Regarding the architecture, the CDB is essentially a standard database which may be implemented using any DBMS. In some situations, an existing database may be used. For this project, the CDB is implemented using dBase III plus. USAISC was analyzed using the SOM methodology described in references [1, 2]. My thanks to Hsiao-Yu Wu for her assistance with this analysis. Based on this analysis, a conceptual design for a database was developed. This conceptual design is depicted in figure 3 at Appendix B. The logical design was then created based on the SOM methodology. The logical design consists of a list of relationships, in fourth normal form, keys, and attributes. Eight relations were required for this project. For the physical design, the relations and keys were implemented using a dBase III plus DBMS. The dBase III plus system is a user friendly product, and the implementation of the database proceeded without difficulty. In order for the MDS/DRS to operate correctly, some joined relations are also required. The database relations are briefly described below and are listed in Appendix A.

Original database relations:

1. ISCTOP Table. This relation contains information about the top (general officer) level of USAISC.
2. ISCDIR Table. This relation contains information about the directorate offices of USAISC.

3. **ISCDIV Table.** This relation contains information about the division offices of USAISC.
4. **ISCBR Table.** This relation contains information about the branch offices of USAISC.
5. **ISCEMP Table.** This relation contains information about the employees of USAISC.
6. **ISCPROJ Table.** This relation contains information about the projects of USAISC.
7. **ISCBRPRO Table.** This relation contains information about the branch offices and their associated projects in USAISC.
8. **ISCEMPPR Table.** This relation contains information about the project officers (employees) and their associated projects of USAISC.

Joined database relations:

1. **DIREMP Table.** This relation contains information resulting from a join about directorate level employees of USAISC.
2. **DVEMP Table.** This relation contains information resulting from a join about division level employees of USAISC.
3. **BREMP Table.** This relation contains information resulting from a join about branch level employees of USAISC.
4. **BRPROCH Table.** This relation contains information resulting from a join about branch chiefs and projects in USAISC.
5. **BDD Table.** This relation contains information resulting from a join about branch and division level employees of USAISC.

## 5.2 Expert System Design

The KMM, and CKB were implemented using the EXSYS Professional expert system (EXSYS). The following sections will describe the implementation of each of these architectural features.

### 5.2.1 Knowledge Management Module (KMM)

The KMM is simply an inference engine which any commercial expert system provides as an integral portion of the product. EXSYS Professional defaults to backward chaining, but it allows the user to select forward chaining. For this project, forward chaining was used by building a command file (called Mailisc.cmd) and placing the commands "forward/f" and "forward/n" in the file. The command file also instructs the expert system to build a report using the Mailisc.out file; the command is "report mailisc.out." The Mailisc.cmd file is located in Appendix A.

### 5.2.2 Corporate Knowledge Base (CKB)

Once the UIM has obtained the required information from the user, it parses the information and places variables understandable to EXSYS Professional into a file called Return.dat. When the batch file calls EXSYS Professional, the program automatically retrieves the information from Return.dat. This information is then inferenced with the rules in the CKB to determine the appropriate queries to send to the CDB. The scenarios and communication policies described in the knowledge acquisition section provide the basis for the rules in the CKB. The conceptual rules are (manually) converted into a set of qualifiers and if-then statements for use in EXSYS Professional. Based on the user responses, the rules determine (this is the "inferencing") what parameters go into a Pass.dat file. The use of the Pass.dat file will be discussed further in the next sections.

## 5.3 User Interface Module (UIM)

Conceptually, the user will enter special keywords at the EMS system "TO" and "SUBJECT" prompts. Given the existence of an EMS system, a small routine will have to assess whether or not a keyword has been entered. Since there is no EMS system for this project, all entries are keywords. Once a keyword has been identified, the UIM obtains the information placed in the "TO" line and the "SUBJECT" line of the message header. This information is then inferenced with the rules in the CKB to determine the appropriate queries to send to the CDB.

The UIM was developed using a batch file, two Pascal programs, and a C program. The batch file, called Go.bat, executes each program in turn. The Pascal program, Mail.pas, presents a "mock" mail system prompt to the user and accepts keyword inputs from the user. Then this routine parses the message header and builds two files: Return.dat and Hdr.dat. The Return.dat file is then accessed by EXSYS Professional; the information provided in Return.dat is inferenced with the rules in the CKB to determine the appropriate queries to send to the CDB. The code for Mail.pas is in Appendix A. After the database query has been processed, the second Pascal program, Combine.pas,

adds header and time limit (when required) information to the list of EMAIL recipients generated by the database query.

The final screen presented to the user was developed by Mike Morrison using the C programming language. It is called report.exe, and it was built for the IOIS project as described earlier. This is an excellent tool for presenting the final list of EMAIL recipients to the user, and it allows the user to ask for help and to scroll through the list of names.

#### **5.4 Corporate Database Interface Module (CDIM)**

The CDIM is a coupling mechanism which interfaces the expert system and the database management system. The CDIM consists of four components:

1. **Go.bat** This is a batch file, the purpose of which is to transfer control among different programs and files. Primarily, it is responsible for switching the EXSYS Professional software and the Query.prg (an executable dBase III plus command file) in and out of main memory. This is necessary because both programs together are too large to fit in 640 kilobytes of main memory simultaneously.
2. **Mailisc.out** This is an EXSYS Professional ASCII file which was created especially for this system. It is the report generator, or format, for the output of the EXSYS Professional session. For this situation, the ".out" file is used to place all of the variables in a special format into the Pass.dat file. The special format for the Pass.dat is required so that the Query.prg program may easily extract the variables it requires from Pass.dat. This process is explained in the next section.
3. **Query.prg** This is an executable dBase III plus command file. The file was written in dBase III plus programming language and compiled using a CLIPPER compiler. This program was developed to work on the special format of data in Pass.dat. It accesses the data in Pass.dat and builds the appropriate queries to get the listing of recipients for that message. The recipient listing is placed in a file called Report.rpt. The executable dBase III plus command file actually includes all of the information stored in the dBase III plus DBMS, and for this reason, it is an extremely large file. My thanks to Milam Aiken and Chih-Ping Wei for their assistance with the Query.prg program.
4. **Combine.pas** This is a Pascal program which uses the ASCII file created by Query.prg, Report.rpt, along with information in the Hdr.dat file (created by Mail.pas) to build a new file called Newrep.rpt. Newrep.rpt contains header information from the original message, the list of EMAIL recipients, and time limits (when required).

Copies of these components are located at Appendix A. The way the different programs are interfaced is presented in the next section.

## 5.5 Process Example

An example of this process will be described using one rule. For the purpose of this project, this entire process is initiated by a batch file called Go.bat. After, the user types in "go," the following actions occur. For a complete list of rules, see Appendix A.

1. At the mail "TO" prompt, the user types in **BRANCH CHIEFS/ASPL-A**. At the mail "SUBJECT" prompt, the user types in **WEEKLY MEETING**. Because of the limited capability of the parser (a Pascal program), the user must restrict his/her input to known keywords.
2. The Mail.pas program (a Pascal program) then performs two actions:
  - (a) First, it parses the input and places three values in a file called Hdr.dat. For this example, the values are:
    - **BRANCH CHIEFS**. These are the recipients, with the sender's division truncated.
    - **WEEKLY MEETING**. This is the subject.
    - **4**. This number is based on the recipient type. The Combine.pas program performs additional actions depending on this number. This will be explained in more detail later.
  - (b) Second, the Mail.pas program translates the user input into qualifier and variable values that EXSYS Professional can understand. Then it places between two and four values in a file called Return.dat. For this example, the values are:
    - **Q1 4**
    - **V12 ASPL-A**
    - **V14 ASPL-A**
    - **Q2 4**
3. Then the batch file calls EXSYS Professional. The user never interacts directly with EXSYS Professional. Instead, EXSYS Professional retrieves all of the information that it needs from the Return.dat file. The data provided in Return.dat tells EXSYS Professional which values for qualifiers and variables have been selected. In this example, value 4 for qualifier 1 (Q1) has been selected. The value of variables 12 and 14 (V12 and V14) is "ASPL-A." The value of qualifier 2 (Q2) is 4.



4. These values cause the fifth rule to fire. This rule is: If the recipients are branch chiefs and the subject is weekly meeting or suspense action then
  - TNAME1 is given the value "BREMP" (BREMP is a joined relation in the database).
  - RFLD1 is given the value "EMP\_ID" (This is the first return field. EMP\_ID is an attribute in the table BREMP which stands for employee id, i.e. the employee's computer name).
  - RFLD2 is given the value "TITLE" (This is the second return field. TITLE is an attribute in the table BREMP which stands for an employee's title, i.e. chief or secretary).
  - MFLD1 is given the value "TITLE" (This is the first matching field).
  - MVALU1 is given the value "CHIEF" (This is the first matching value).
  - MFLD3 is given the value "DIV\_ABBR" (This is the second matching field).
  - MVALU3 is given the value "ASPL-A" (This is the second matching value. It is provided by Return.dat, i.e. V12 = ASPL-A).
  - TNAME2 is given the value "BREMP."
  - RFLD3 is given the value "EMP\_ID."
  - RFLD4 is given the value "TITLE."
  - MFLD2 is given the value "TITLE."
  - MVALU2 is given the value "SECY."
  - MFLD4 is given the value "DIV\_ABBR."
  - MVALU4 is given the value "ASPL-A." (This is also provided by Return.dat, i.e. V14 = ASPL-A).
5. EMP\_ID and TITLE are attributes in the joined relation called BREMP (branch employees). The above information is placed in a file called Pass.dat.
6. The batch file takes EXSYS Professional out of main memory and places the dBase III plus executable file (Query.exe) in main memory. Query.exe uses the above information to build a query and return the appropriate EMP\_IDs and TITLES. The above information essentially tells Query.prg to build a query which will go to the table called BREMP and select the EMP\_ID (RFLD1) and TITLE (RFLD2) of every employee whose TITLE (MFLD1) is CHIEF (MVALU1) and whose DIV\_ABBR (MFLD3) (division abbreviation) is ASPL-A (MVALU3). The process is repeated for the second portion of the information presented above to find EMP\_IDs and TITLES for the secretaries.
7. The listing of recipients is placed in an ASCII file called Report.rpt. Control returns to the batch file, and it executes the Pascal program Combine.exe.

8. Combine.exe combines the information in Hdr.dat and Report.rpt to create an ASCII file called Newrep.rpt. Combine also tests the number provided in Hdr.dat. If the number is within the range of 8, 9, or 10, this indicates that it is a document routing request. For document routing, Combine will also place a time limit of one day next to the name of each chief in the list. This is not necessarily a realistic time limit, but it demonstrates the concept that time limits may be placed on the recipients of documents that are being routed up through the chain of command. Control returns to the batch file, and it executes the C program Report.exe.
9. Report.exe then accesses the file Newrep.rpt and displays the information contained in Newrep.rpt.

## 6 Future Directions

This project has demonstrated that a KMS providing MDS and DRS features is a feasible task that provides advantages to the user. However, there are several issues which should be considered for future implementations of this or similar systems. These issues are discussed here.

1. **Database Interface.** EXSYS Professional (version 1.1.4) claims to provide a database interface; however, this interface is inefficient because it accesses a table in the database one record at a time. This requires a significant amount of communication between EXSYS Professional and dBase III plus. Additionally, if one or both of the programs are too large, they will exceed the main memory capacity (640 kilobytes) of a PC. Later versions of EXSYS Professional, as well as add-on third party software, may obviate this problem.
2. **Query.prg.** This program currently supports only a very simple dBase III plus database. For example, Query.prg expects every attribute to be of the same type and length. For this project every attribute in the dBase III plus database is of type character and has a length of 40. Of course, this is unrealistic for supporting normal databases which would allow for attributes of all types and lengths. This program would probably have to be tailored to "fit" any existing database of an organization.
3. **EMS Interface.** Eventually, in accordance with the KMS architecture, the MDS and DRS developed for this project would need to be interfaced with an EMS system. This would allow the user to omit his/her branch, division, or directorate designation. In the example provided earlier, the user had to add "ASPL-A" to designate the division of the branch chiefs. The EMS could automatically provide this information to EXSYS Professional, relieving the user of this burden. For example,

the division chief of ASPL-A should be able to type in "BRANCH CHIEFS" and "WEEKLY MEETING," with the understanding that the EMS would supply the sender's "ASPL-A" designation to EXSYS Professional. This would also reduce some of the work of the parser program, i.e. it would no longer have to parse the "TO" line into a sender and a receiver.

4. **Error Correction.** There are no facilities for error correction in this program. Error correction would have to be included in the UIM to ensure successful operation of the system, no matter what the user input. Two types of error correction are necessary:
  - (a) *Correct Keyword Input.* If the user provides misspelled or otherwise incorrect keywords, the system will either have to prompt the user for correct keywords or provide help with the correct keywords. One possibility is to take advantage of the inherent menu feature of EXSYS Professional. Right now, this feature is effectively turned off by the command "nooutput" in the Mailisc.cfg file. This was done to provide an embedded system. However, under normal circumstances, EXSYS Professional will prompt the user for correct input, even if incorrect input is initially provided in a Return.dat file. Another possibility is to provide a list of keywords from which the user may choose. This list could be produced by the Pascal program which accepts input from the user.
  - (b) *Uppercase Input.* The input provided to EXSYS Professional must be in uppercase letters. This kind of error correction can be easily provided by the Pascal program which accepts the input from the user.
5. **Parser Program.** The parser program built for this project is very small and limited in its ability to parse. As parsing techniques improve, this portion of the program could be significantly enhanced. The system should be able to pick out keywords from any phrase, i.e. to parse "meeting" out of the phrase "special meeting on new technology."
6. **UIM/KMM/CKB Interface.** The UIM/KMM/CKB interface for this system is very highly coupled. This means that when the CKB is updated with new rules, the UIM has to be updated as well. A loosely coupled interface would provide for easier maintenance of the UIM as well as the CKB. Currently, assignment of EXSYS Professional variables and qualifiers occurs in the Pascal program Mail.exe, which is also the UIM. In the future, I recommend that the UIM be primarily a parsing program, as described above. The UIM should parse and send the keyword to EXSYS Professional, and then rules in EXSYS Professional should assign variables and qualifiers accordingly. This would greatly simplify maintenance of the UIM and the CKB.

7. **Explanation Feature.** This system provides very little assistance to the user. "Help" features would increase the user-friendliness of this system. There are two possibilities for implementing explanation facilities:
- (a) The first method of incorporating help features is to include an additional help program in the system which could be accessed through the UIM (Mail.exe). This facility would need to assist the user in deciding which keywords are needed to generate the appropriate list of recipients. It may be appropriate for this feature to be called automatically upon failure of the parsing program to discover a keyword.
  - (b) The second approach involves incorporating the help facility into the CKB. In the event that the UIM could not provide a keyword to the CKB, then the CKB would be triggered to step the user through a series of menus in order to determine the necessary information to generate a list of recipients. EXSYS Professional is designed to "query" the user for information, so this would be a very natural explanation feature. Additional ASCII help screens can be written to assist the user in understanding the use of EXSYS Professional. The "nooutput" command in the Mailisc.cfg file would have to be removed in order for the user to access EXSYS Professional in this fashion.
8. **Validation Feature.** A validation feature will have to be added to allow the user to add to, delete from, or verify the list of potential recipients before transmission. The validation feature would probably be a part of the UIM or the EIM or both.
9. **Joined Relations.** Currently, joined relations required for the correct operation of the system are joined ahead of time and essentially are a static portion of the system. In a real world system, the tables would need to be dynamically joined at the beginning of the Mail.pas program which accepts keyword input from the user. Another option would be for the system to join the relations periodically, perhaps every three hours and to use that information. This would not be completely current information, but it would probably be adequate.
10. **Training and User Acceptance.** EMS systems are currently used by action officers in USAISC for communication with external commands. They are rarely used for internal communication. The MDS/DRS would require new learning and acceptance on the part of the workers of SAISC. The chain of command would need to provide training for the users regarding these new facilities.
11. **Intra-Organizational Communication.** This system is concerned with intra-organizational communication, i.e. communication within USAISC. Given a database which included the appropriate information from other Army commands (external to USAISC), rules could be developed to disseminate messages on a much larger scale, i.e. to affect inter-organizational communication. Many more database issues

would need to be addressed before this system could be implemented. For example, which organization would maintain the database; who would ensure accurate input, etc.

12. **CKB Expansion.** This system was developed primarily to deal with project topics and some administrative topics. However, there are applications for other areas within the purview of an Army command. For example, personnel offices communicate frequently with a given group of individuals. The CKB could be expanded to include rules for this kind of communication as well. Two possibilities are presented:

- The following policy is given:
  - Whenever officers of the rank Captain are notified of generic personnel information, their immediate supervisors will also be notified. This type of a policy permits supervisors to counsel their subordinates regarding the importance of the information.

For example, the personnel office may need to disseminate the following information: All Captains of year group '81 need to submit a photograph to the Department of the Army prior to September 1990 because the promotion board will convene in September 1990. In this situation, the MDS would build a list consisting of all Captains in year group '81 and their supervisors (probably branch chiefs). The rule could be stated in the following generic terms: Find all officers in the database that are captains and are in year group '81, and find their supervisors. Ideally, the supervisors would discuss with their subordinates the importance (for promotion purposes) of submitting the photograph on time. In order to support queries of this nature, the database would have to contain a significant amount of personnel information. An alternative is to have a separate personnel database (many organizations already have these) which is used to support these kind of queries. However, this will increase the complexity of the interface mechanism (CDIM).

- Another policy may be:
  - Whenever there is a policy change regarding the wear of the officer uniform, notify all officers of that change. This policy immediately informs all officers, supervisory or subordinate, of the change.

For example, the personnel office may need to disseminate the following information: As of 1 December 1985, female officers will wear branch and U.S. insignia on both lapels of the green jacket. For this scenario, the MDS would build a list consisting of all officers in the database. Stated in generic terms, the rule is: Find all officers in the database. This policy informs both supervisors and subordinates of the new uniform change.

## 7 Conclusion

There are many advantages to be gained by incorporating intelligence into the use of a standard EMS. The KMS architecture provides the framework for building an intelligent EMS which takes advantage of existing corporate knowledge. As organizations increase their use of electronic mail, any feature which increases the efficiency of communicating through this medium will gain in popularity. Now, some of the specific advantages of the MDS and the DRS will be examined.

### 7.1 Advantages of an MDS

An MDS provides support for organizational communication to managers and workers. The advantages of using an MDS are:

1. It determines who among the staff need to receive a particular message. The sender is not required to determine that for himself/herself. Potentially, this could save the user a considerable amount of time.
2. It prevents "junk mail" in that staff members not involved in the message topic do not receive the message. Reading and removing junk mail can be a time-consuming process, so by preventing junk mail, the efficiency of the organization may be improved.
3. The MDS encourages communication among organizational personnel in an orderly fashion. In so doing, it discourages situations where personnel in one section of the organization do not know what other personnel are doing. If the database is maintained properly, personnel should be kept informed of all activities in the organization based on their projects. To some extent, the users may experience an increase in the amount of mail they receive; however, it should not be "junk mail."

### 7.2 Advantages of a DRS

A DRS also provides support for organizational communication, specifically for those documents which must be routed upward through a chain of command.

1. The DRS determines who the recipients are based on the level through which the document must be routed and provides an ordered list of these recipients. The sender is not required to determine that for himself/herself. Certainly, the sender would know his/her own chain of command for routing purposes; however, considerable time is saved by having this done electronically instead of manually.

The sender need not constantly monitor (telephonically or otherwise) the progress of the document.

2. The DRS may also provide time limits for each stop for the document. In this manner, the document may proceed without delay. This results in additional time saved for the organization.

## 8 Appendix A

### 8.1 Database Schema

This is a listing of the database schema. It includes the original relations as well as the joined relations required for this project.

### 8.2 Rules Listing

This is a listing of the rules for the EXSYS Professional expert system. It also includes a listing of the qualifiers and variables used by EXSYS Professional.

### 8.3 Go.bat

Go.bat is the batch file which switches the different programs in and out of main memory.

### 8.4 Mailisc.cmd

The .cmd file may be used to control the direction of inferencing. For example, the "forward/f" command in this file specifies that forward chaining will be used, and the "forward/n" command specifies that no backward chaining will be used. The command "report mailisc.out" tells the expert system to generate a report using the format provided by Mailisc.out.

### 8.5 Mailisc.out

A .out file is the file which formats the data passed into the Pass.dat file by EXSYS Professional.

### 8.6 Mailisc.cfg

A .cfg (configuration) file allows the user to specify commands which apply to the execution of EXSYS Professional. The "datalist" command tells EXSYS Professional to read the Return.dat file for input information. The "nooutput" command prevents any output during the execution of EXSYS Professional; hence, the appearance of an "embedded" system.



### **8.7 Mail.pas**

This is a Pascal program which provides a mock user interface; it appears as though the user were in an electronic mail system. It also receives and parses the user input.

### **8.8 Combine.pas**

This is the Pascal program which adds the original header information on to the list of EMAIL recipients before it is send to the report display program, Report.exe. The name of the output of this program is Newrep.rpt.

### **8.9 Newrep.hlp**

This is an ASCII file which must be present in order for the Report.exe file to operate correctly.

### **8.10 Query.prg**

This is the dBase III plus program that takes input from Pass.dat and creates output for Report.txt.

### **8.11 Read.me**

This is a file which describes and lists the files and programs required to run the MDS and DRS. It also tells the user how to start the program.

Appendix A

**DATABASE SCHEMA**

Database schema:

Entity Name: ISCTOP  
Identifier: Top\_Abbr  
Properties: Lev\_Name  
Phone

Foreign Key:

Entity Name: ISCDIR  
Identifier: Dir\_Abbr  
Properties: Lev\_Name  
Phone

Foreign Key: Top\_Abbr

Entity Name: ISCDIV  
Identifier: Div\_Abbr  
Properties: Lev\_Name  
Phone

Foreign Key: Dir\_Abbr

Entity Name: ISCBR  
Identifier: Br\_Abbr  
Properties: Lev\_Name  
Phone

Foreign Key: Div\_Abbr

Entity Name: ISCEMP  
Identifier: Emp\_Id  
Properties: Emp\_Name  
Title

Foreign Key: Lev\_Abbr

Entity Name: ISCPROJ  
Identifier: ProjId  
Properties: Proj\_Name  
Budget  
Status

Foreign Key:

Entity Name: ISCBRPRO  
Identifier: Br\_Abbr  
ProjId

Properties: Primary  
Foreign Key: Br\_Abbr  
ProjId

Entity Name: ISCEMPPR  
Identifier: ProjId  
Emp\_Id

Properties: Primary  
Foreign Key: ProjId  
Emp\_Id

Database Schema for Joined Tables:

Entity Name: DIREMP

Tables Joined: ISCDIR  
ISCEMP  
Properties: Emp\_Id  
Emp\_Name  
Title  
Lev\_Abbr

Entity Name: DVEMP  
Tables Joined: ISCDIV  
ISCEMP  
Properties: Emp\_Id  
Title  
Lev\_Abbr  
Dir\_Abbr

Entity Name: BREMP  
Tables Joined: ISCBR  
ISCEMP  
Properties: Emp\_Id  
Title  
Lev\_Abbr  
Div\_Abbr

Entity Name: BRPROCH  
Tables Joined: ISCBRPRO  
ISCEMP  
Properties: Emp\_Id  
Title  
Br\_Abbr  
ProjId  
Primary

Entity Name: BDD  
Tables Joined: ISCDIV  
BREMP  
Properties: Emp\_Id  
Title  
Lev\_Abbr  
Div\_Abbr  
Dir\_Abbr

Appendix A

**RULES LISTING**

LIST ONLY

Subject:

This expert system will select mail users given a subject line and a to line.

Author:

Pamela O. Howard

Starting text:

This will be an embedded system.

Ending text:

Hopefully, this system will be embedded, and this text will not be seen.

Derivation: ALL RULES USED

PROBABILITY SYSTEM: 1

DISPLAY THRESHOLD: 1

QUALIFIERS:

/\* Qualifier 1

Q> THE RECIPIENTS ARE

V> PRIMARY PROJECT OFFICERS

V> DIVISION & BRANCH CHIEFS

V> DIVISION CHIEFS

V> BRANCH CHIEFS

V> DIRECTORATE CHIEFS

V> PROJECT OFFICERS

V> ALL EMPLOYEES

V> DOC RTNG - DIV

V> DOC RTNG - DIR

V> DOC RTNG - GENERAL

/\* Qualifier 2

Q> THE SUBJECT IS

V> FUNDING MEETING  
V> TRAINING COURSES  
V> MEETING  
V> WEEKLY MEETING  
V> PRIVATE  
V> SUSPENSE ACTION  
V> OTHER  
V> FYI  
V> DOCUMENT APPROVAL  
V> WEEKLY ACTIVITY REPORT  
V> INFO PAPER  
V> MEETING TO ATTEND  
V> BLUE BULLET

CHOICES:

VARIABLES:

[TNAME1] FIRST TABLE NAME  
Type = S

[RFLD1] FIRST RETURN FIELD  
Type = S

[RFLD2] SECOND RETURN FIELD  
Type = S

[MFLD1] FIRST MATCHING FIELD  
Type = S

[MVALU1] FIRST MATCHING VALUE  
Type = S

[TNAME2] SECOND TABLE NAME  
Type = S

[RFLD3] THIRD RETURN FIELD  
Type = S

[RFLD4] FOURTH RETURN FIELD  
Type = S

[MFLD2] SECOND MATCHING FIELD  
Type = S

[MVALU2] SECOND MATCHING VALUE  
Type = S

[MFLD3] THIRD MATCHING FIELD  
Type = S

[MVALU3] THIRD MATCHING VALUE  
Type = S

[MFLD4] FOURTH MATCHING FIELD  
Type = S

[MVALU4] FOURTH MATCHING VALUE  
Type = S

[TNAME3] THIRD TABLE NAME  
Type = S

[RFLD5] FIFTH RETURN FIELD  
Type = S

[RFLD6] SIXTH RETURN FIELD  
Type = S

[MFLD5] FIFTH MATCHING FIELD  
Type = S

[MVALU5] FIFTH MATCHING VALUE  
Type = S

[MFLD6] SIXTH MATCHING FIELD  
Type = S

[MVALU6] SIXTH MATCHING VALUE  
Type = S

[TNAME4] FOURTH TABLE NAME  
Type = S

[RFLD7] SEVENTH RETURN FIELD  
Type = S

[RFLD8] EIGHTH RETURN FIELD  
Type = S



[MFLD7] SEVENTH MATCHING FIELD  
Type = S

[MVALU7] SEVENTH MATCHING VALUE  
Type = S

[MFLD8] EIGHTH MATCHING FIELD  
Type = S

[MVALU8] EIGHTH MATCHING VALUE  
Type = S

[TNAME5] FIFTH TABLE NAME  
Type = S

[RFLD9] NINTH RETURN FIELD  
Type = S

[RFLD10] TENTH RETURN FIELD  
Type = S

[MFLD9] NINTH MATCHING FIELD  
Type = S

[MVALU9] NINTH MATCHING VALUE  
Type = S

[MFLD10] TENTH MATCHING FIELD  
Type = S

[MVALU10] TENTH MATCHING VALUE  
Type = S

[TNAME6] SIXTH TABLE NAME  
Type = S

[RFLD11] RETURN FIELD 11  
Type = S

[RFLD12] RETURN FIELD 12  
Type = S

[MFLD11] MATCHING FIELD 11  
Type = S

[MVALU11] MATCHING VALUE 11  
Type = S

[MFLD12] MATCHING FIELD 12  
Type = S

[MVALU12] MATCHING VALUE 12  
Type = S

[TNAME7] TABLE NAME 7  
Type = S

[RFLD13] RETURN FIELD 13  
Type = S

[RFLD14] RETURN FIELD 14  
Type = S

[MFLD13] MATCHING FIELD 13  
Type = S

[MVALU13] MATCHING VALUE 13  
Type = S

[MFLD14] MATCHING FIELD 14  
Type = S

[MVALU14] MATCHING VALUE 14  
Type = S

[TNAME8] TABLE NAME 8  
Type = S

[RFLD15] RETURN FIELD 15  
Type = S

[RFLD16] RETURN FIELD 16  
Type = S

[MFLD15] MATCHING FIELD 15  
Type = S

[MVALU15] MATCHING VALUE 15  
Type = S

[MFLD16] MATCHING FIELD 16  
Type = S

[MVALU16] MATCHING VALUE 16  
Type = S

RULES:

/\* RULE NUMBER: 1

IF:

THE RECIPIENTS ARE {PRIMARY PROJECT OFFICERS}  
and: THE SUBJECT IS {FUNDING MEETING}

THEN:

[TNAME1] IS GIVEN THE VALUE "ISCEMPPR"  
and: [RFLD1] IS GIVEN THE VALUE "EMP\_ID"  
and: [RFLD2] IS GIVEN THE VALUE "PROJID"  
and: [MFLD1] IS GIVEN THE VALUE "PRIMARY"  
and: [MVALU1] IS GIVEN THE VALUE "TRUE"  
and: [TNAME2] IS GIVEN THE VALUE "BRPROCH"  
and: [RFLD3] IS GIVEN THE VALUE "EMP\_ID"  
and: [RFLD4] IS GIVEN THE VALUE "PROJID"  
and: [MFLD2] IS GIVEN THE VALUE "TITLE"  
and: [MVALU2] IS GIVEN THE VALUE "CHIEF"  
and: [MFLD4] IS GIVEN THE VALUE "PRIMARY"  
and: [MVALU4] IS GIVEN THE VALUE "TRUE"

/\* RULE NUMBER: 2

IF:

THE RECIPIENTS ARE {DIVISION & BRANCH CHIEFS}  
and: THE SUBJECT IS {TRAINING COURSES}

THEN:

[TNAME1] IS GIVEN THE VALUE "DVEMP"  
and: [RFLD1] IS GIVEN THE VALUE "EMP\_ID"  
and: [RFLD2] IS GIVEN THE VALUE "TITLE"  
and: [MFLD1] IS GIVEN THE VALUE "TITLE"  
and: [MVALU1] IS GIVEN THE VALUE "CHIEF"  
and: [MFLD3] IS GIVEN THE VALUE "DIR\_ABBR"  
and: [MVALU3] IS GIVEN THE VALUE [MVALU3]  
and: [TNAME2] IS GIVEN THE VALUE "BDD"  
and: [RFLD3] IS GIVEN THE VALUE "EMP\_ID"  
and: [RFLD4] IS GIVEN THE VALUE "TITLE"  
and: [MFLD2] IS GIVEN THE VALUE "TITLE"  
and: [MVALU2] IS GIVEN THE VALUE "CHIEF"  
and: [MFLD4] IS GIVEN THE VALUE "DIR\_ABBR"  
and: [MVALU4] IS GIVEN THE VALUE [MVALU4]

/\* RULE NUMBER: 3

IF:

THE RECIPIENTS ARE {DIVISION CHIEFS}  
and: THE SUBJECT IS {MEETING}

THEN:

[TNAME1] IS GIVEN THE VALUE "DVEMP"  
and: [RFLD1] IS GIVEN THE VALUE "EMP\_ID"  
and: [RFLD2] IS GIVEN THE VALUE "TITLE"  
and: [MFLD1] IS GIVEN THE VALUE "TITLE"  
and: [MVALU1] IS GIVEN THE VALUE "CHIEF"  
and: [MFLD3] IS GIVEN THE VALUE "DIR\_ABBR"  
and: [MVALU3] IS GIVEN THE VALUE [MVALU3]  
and: [TNAME2] IS GIVEN THE VALUE "DVEMP"  
and: [RFLD3] IS GIVEN THE VALUE "EMP\_ID"  
and: [RFLD4] IS GIVEN THE VALUE "TITLE"  
and: [MFLD2] IS GIVEN THE VALUE "TITLE"  
and: [MVALU2] IS GIVEN THE VALUE "SECY"  
and: [MFLD4] IS GIVEN THE VALUE "DIR\_ABBR"  
and: [MVALU4] IS GIVEN THE VALUE [MVALU4]

/\* RULE NUMBER: 4

IF:

THE RECIPIENTS ARE {DIVISION CHIEFS}  
and: THE SUBJECT IS {OTHER}

THEN:

[TNAME1] IS GIVEN THE VALUE "DVEMP"  
and: [RFLD1] IS GIVEN THE VALUE "EMP\_ID"  
and: [RFLD2] IS GIVEN THE VALUE "TITLE"  
and: [MFLD1] IS GIVEN THE VALUE "TITLE"  
and: [MVALU1] IS GIVEN THE VALUE "CHIEF"  
and: [MFLD3] IS GIVEN THE VALUE "DIR\_ABBR"  
and: [MVALU3] IS GIVEN THE VALUE [MVALU3]

/\* RULE NUMBER: 5

IF:

THE RECIPIENTS ARE {BRANCH CHIEFS}  
and: THE SUBJECT IS {WEEKLY MEETING} OR {SUSPENSE ACTION}

THEN:

[TNAME1] IS GIVEN THE VALUE "BREMP"  
and: [RFLD1] IS GIVEN THE VALUE "EMP\_ID"  
and: [RFLD2] IS GIVEN THE VALUE "TITLE"  
and: [MFLD1] IS GIVEN THE VALUE "TITLE"  
and: [MVALU1] IS GIVEN THE VALUE "CHIEF"  
and: [MFLD3] IS GIVEN THE VALUE "DIV\_ABBR"  
and: [MVALU3] IS GIVEN THE VALUE [MVALU3]  
and: [TNAME2] IS GIVEN THE VALUE "BREMP"

and: [RFLD3] IS GIVEN THE VALUE "EMP\_ID"  
and: [RFLD4] IS GIVEN THE VALUE "TITLE"  
and: [MFLD2] IS GIVEN THE VALUE "TITLE"  
and: [MVALU2] IS GIVEN THE VALUE "SECY"  
and: [MFLD4] IS GIVEN THE VALUE "DIV\_ABBR"  
and: [MVALU4] IS GIVEN THE VALUE [MVALU4]

/\* RULE NUMBER: 6

IF:  
THE RECIPIENTS ARE {DIRECTORATE CHIEFS}  
and: THE SUBJECT IS {PRIVATE}

THEN:  
[TNAME1] IS GIVEN THE VALUE "DIREMP"  
and: [RFLD1] IS GIVEN THE VALUE "EMP\_ID"  
and: [RFLD2] IS GIVEN THE VALUE "LEV\_ABBR"  
and: [MFLD1] IS GIVEN THE VALUE "TITLE"  
and: [MVALU1] IS GIVEN THE VALUE "CHIEF"

/\* RULE NUMBER: 7

IF:  
THE RECIPIENTS ARE {DIRECTORATE CHIEFS}  
and: THE SUBJECT IS {OTHER}

THEN:  
[TNAME1] IS GIVEN THE VALUE "DIREMP"  
and: [RFLD1] IS GIVEN THE VALUE "EMP\_ID"  
and: [RFLD2] IS GIVEN THE VALUE "LEV\_ABBR"  
and: [MFLD1] IS GIVEN THE VALUE "TITLE"  
and: [MVALU1] IS GIVEN THE VALUE "CHIEF"  
and: [TNAME2] IS GIVEN THE VALUE "DIREMP"  
and: [RFLD3] IS GIVEN THE VALUE "EMP\_ID"  
and: [RFLD4] IS GIVEN THE VALUE "LEV\_ABBR"  
and: [MFLD2] IS GIVEN THE VALUE "TITLE"  
and: [MVALU2] IS GIVEN THE VALUE "SECY"

/\* RULE NUMBER: 8

IF:  
THE RECIPIENTS ARE {PROJECT OFFICERS}

THEN:  
[TNAME1] IS GIVEN THE VALUE "ISCEMPPR"  
and: [RFLD1] IS GIVEN THE VALUE "EMP\_ID"  
and: [RFLD2] IS GIVEN THE VALUE "PROJID"  
and: [MFLD1] IS GIVEN THE VALUE "PROJID"  
and: [MVALU1] IS GIVEN THE VALUE [MVALU1]  
and: [TNAME2] IS GIVEN THE VALUE "BRPROCH"  
and: [RFLD3] IS GIVEN THE VALUE "EMP\_ID"

and: [RFLD4] IS GIVEN THE VALUE "TITLE"  
and: [MFLD2] IS GIVEN THE VALUE "TITLE"  
and: [MVALU2] IS GIVEN THE VALUE "CHIEF"  
and: [MFLD4] IS GIVEN THE VALUE "PROJID"  
and: [MVALU4] IS GIVEN THE VALUE [MVALU4]

/\* RULE NUMBER: 9

IF:  
    THE RECIPIENTS ARE {ALL EMPLOYEES}  
  
THEN:  
    [TNAME1] IS GIVEN THE VALUE "ISCEMP"  
and: [RFLD1] IS GIVEN THE VALUE "EMP\_ID"  
and: [RFLD2] IS GIVEN THE VALUE "EMP\_NAME"

/\* RULE NUMBER: 10

IF:  
    THE RECIPIENTS ARE {DOC RTNG - DIV} OR {DOC RTNG - DIR} OR {DOC RTNG -  
    GENERAL}  
and: [MVALU3] = "ASPL-AA" OR [MVALU3] = "ASPL-AL" OR [MVALU3] = "ASPL-AT"  
  
THEN:  
    [MVALU6] IS GIVEN THE VALUE "ASPL-A"

/\* RULE NUMBER: 11

IF:  
    THE RECIPIENTS ARE {DOC RTNG - DIV} OR {DOC RTNG - DIR} OR {DOC RTNG -  
    GENERAL}  
and: [MVALU3] = "ASRM-PBI" OR [MVALU3] = "ASRM-PBM" OR [MVALU3] =  
    "ASRM-PBP"  
  
THEN:  
    [MVALU6] IS GIVEN THE VALUE "ASRM-PB"

/\* RULE NUMBER: 12

IF:  
    [MVALU6] = "ASPL-A" OR [MVALU6] = "ASPL-I" OR [MVALU6] = "ASPL-P"  
and: THE RECIPIENTS ARE {DOC RTNG - DIR} OR {DOC RTNG - GENERAL}  
  
THEN:  
    [MVALU10] IS GIVEN THE VALUE "ASPL"

/\* RULE NUMBER: 13

IF:

[MVALU6] = "ASRM-F" OR [MVALU6] = "ASRM-M" OR [MVALU6] = "ASRM-CA" OR  
[MVALU6] = "ASRM-CT" OR [MVALU6] = "ASRM-PB"

and: THE RECIPIENTS ARE {DOC RTNG - DIR} OR {DOC RTNG - GENERAL}

THEN:

[MVALU10] IS GIVEN THE VALUE "ASRM"

/\* RULE NUMBER: 14

IF:

THE RECIPIENTS ARE {DOC RTNG - GENERAL}

and: [MVALU10] = "ASPL" OR [MVALU10] = "ASRM" OR [MVALU10] = "ASOP" OR  
[MVALU10] = "ASLO" OR [MVALU10] = "ASIM" OR [MVALU10] = "ASIS"  
OR [MVALU10] = "ASPE"

THEN:

[MVALU14] IS GIVEN THE VALUE "ASCG"

/\* RULE NUMBER: 15

IF:

THE RECIPIENTS ARE {DOC RTNG - DIV} OR {DOC RTNG - DIR} OR {DOC RTNG - GENERAL}

and: THE SUBJECT IS {OTHER} OR {FYI} OR {DOCUMENT APPROVAL} OR {WEEKLY  
ACTIVITY REPORT} OR {INFO PAPER} OR {MEETING TO ATTEND} OR {BLUE  
BULLET}

THEN:

[TNAME1] IS GIVEN THE VALUE "BREMP"  
and: [RFLD1] IS GIVEN THE VALUE "EMP\_ID"  
and: [RFLD2] IS GIVEN THE VALUE "TITLE"  
and: [MFLD1] IS GIVEN THE VALUE "TITLE"  
and: [MVALU1] IS GIVEN THE VALUE "CHIEF"  
and: [MFLD3] IS GIVEN THE VALUE "LEV\_ABBR"  
and: [MVALU3] IS GIVEN THE VALUE [MVALU3]  
and: [TNAME2] IS GIVEN THE VALUE "BREMP"  
and: [RFLD3] IS GIVEN THE VALUE "EMP\_ID"  
and: [RFLD4] IS GIVEN THE VALUE "TITLE"  
and: [MFLD2] IS GIVEN THE VALUE "TITLE"  
and: [MVALU2] IS GIVEN THE VALUE "SECY"  
and: [MFLD4] IS GIVEN THE VALUE "LEV\_ABBR"  
and: [MVALU4] IS GIVEN THE VALUE [MVALU3]  
and: [TNAME3] IS GIVEN THE VALUE "DVEMP"  
and: [RFLD5] IS GIVEN THE VALUE "EMP\_ID"  
and: [RFLD6] IS GIVEN THE VALUE "TITLE"  
and: [MFLD5] IS GIVEN THE VALUE "TITLE"  
and: [MVALU5] IS GIVEN THE VALUE "CHIEF"  
and: [MFLD6] IS GIVEN THE VALUE "LEV\_ABBR"  
and: [MVALU6] IS GIVEN THE VALUE [MVALU6]  
and: [TNAME4] IS GIVEN THE VALUE "DVEMP"

and: [RFLD7] IS GIVEN THE VALUE "EMP\_ID"  
and: [RFLD8] IS GIVEN THE VALUE "TITLE"  
and: [MFLD7] IS GIVEN THE VALUE "TITLE"  
and: [MVALU7] IS GIVEN THE VALUE "SECY"  
and: [MFLD8] IS GIVEN THE VALUE "LEV\_ABBR"  
and: [MVALU8] IS GIVEN THE VALUE [MVALU6]

/\* RULE NUMBER: 16

IF:

THE RECIPIENTS ARE {DOC RTNG - DIR} OR {DOC RTNG - GENERAL}  
and: THE SUBJECT IS {OTHER} OR {FYI} OR {DOCUMENT APPROVAL} OR {WEEKLY  
ACTIVITY REPORT} OR {INFO PAPER} OR {MEETING TO ATTEND} OR {BLUE  
BULLET}

THEN:

[TNAME5] IS GIVEN THE VALUE "DIREMP"  
and: [RFLD9] IS GIVEN THE VALUE "EMP\_ID"  
and: [RFLD10] IS GIVEN THE VALUE "TITLE"  
and: [MFLD9] IS GIVEN THE VALUE "TITLE"  
and: [MVALU9] IS GIVEN THE VALUE "CHIEF"  
and: [MFLD10] IS GIVEN THE VALUE "LEV\_ABBR"  
and: [MVALU10] IS GIVEN THE VALUE [MVALU10]  
and: [TNAME6] IS GIVEN THE VALUE "DIREMP"  
and: [RFLD11] IS GIVEN THE VALUE "EMP\_ID"  
and: [RFLD12] IS GIVEN THE VALUE "TITLE"  
and: [MFLD11] IS GIVEN THE VALUE "TITLE"  
and: [MVALU11] IS GIVEN THE VALUE "SECY"  
and: [MFLD12] IS GIVEN THE VALUE "LEV\_ABBR"  
and: [MVALU12] IS GIVEN THE VALUE [MVALU10]

/\* RULE NUMBER: 17

IF:

THE RECIPIENTS ARE {DOC RTNG - GENERAL}  
and: THE SUBJECT IS {OTHER} OR {INFO PAPER} OR {MEETING TO ATTEND} OR {BLUE  
BULLET}

THEN:

[TNAME7] IS GIVEN THE VALUE "ISCEMP"  
and: [RFLD13] IS GIVEN THE VALUE "EMP\_ID"  
and: [RFLD14] IS GIVEN THE VALUE "TITLE"  
and: [MFLD13] IS GIVEN THE VALUE "TITLE"  
and: [MVALU13] IS GIVEN THE VALUE "CHIEF"  
and: [MFLD14] IS GIVEN THE VALUE "LEV\_ABBR"  
and: [MVALU14] IS GIVEN THE VALUE [MVALU14]  
and: [TNAME8] IS GIVEN THE VALUE "ISCEMP"  
and: [RFLD15] IS GIVEN THE VALUE "EMP\_ID"  
and: [RFLD16] IS GIVEN THE VALUE "TITLE"  
and: [MFLD15] IS GIVEN THE VALUE "TITLE"  
and: [MVALU15] IS GIVEN THE VALUE "SECY"  
and: [MFLD16] IS GIVEN THE VALUE "LEV\_ABBR"  
and: [MVALU16] IS GIVEN THE VALUE [MVALU14]



Appendix A

GO.BAT

Go.bat:

ECHO OFF

CLS

REM -----

REM     This is the batch file for the Message Dissemination  
REM     System and the Document Routing System developed by  
REM     Pamela O. Howard for a Masters project, Spring 1990

REM -----

REM     Go to directory with AIMAIL system

REM CD C:\EXSYS\SYS\RUNTIME\PAM

REM     Turn off EXSYS logo

SET EXSYS\_TITLE=(C) EXSYS Inc., 1988

REM     Delete files which pass parameters, if they exist

IF EXIST RETURN.DAT DEL RETURN.DAT

IF EXIST PASS.DAT DEL PASS.DAT

REM     Run Pascal program which parses mail header

MAIL

REM     Run expert system called Mailisc

REM C:\EXSYS\SYS\RUNTIME\EXSYSP.COM MAILISC

EXSYSP MAILISC

REM     Run dBase program called Query to generate distribution list

QUERY

REM     Run pascal program called Combine to add header and timing

REM     information to the list built by Query

COMBINE

REM     Show the distribution list using Mike Morrison's REPORT program

REM     format REPORT.EXE filename.RPT "Header Info"

REPORT NEWREP "Message Dissemination System"

CLS

Appendix A

MAILISC.CMD

Mailisc.cmd:

rules all forward/f forward/n  
report mailisc.out

Appendix A  
MAILISC.OUT

Mailisc.out:

FILE PASS.DAT  
V 1 /V  
V 2 /V  
V 3 /V  
V 4 <> "" /V  
V 5 <> "" /V  
V 11 <> "" /V  
V 12 <> "" /V  
V 1 /"END"  
V 6 /V  
V 7 /V  
V 8 /V  
V 9 /V  
V 10 /V  
V 13 <> "" /V  
V 14 <> "" /V  
V 6 /"END"  
V 15 /V  
V 16 /V  
V 17 /V  
V 18 /V  
V 19 /V  
V 20 <> "" /V  
V 21 <> "" /V  
V 15 /"END"  
V 22 /V  
V 23 /V  
V 24 /V  
V 25 /V  
V 26 /V  
V 27 <> "" /V  
V 28 <> "" /V  
V 22 /"END"  
V 29 /V  
V 30 /V  
V 31 /V  
V 32 /V  
V 33 /V  
V 34 <> "" /V  
V 35 <> "" /V  
V 29 /"END"  
V 36 /V  
V 37 /V  
V 38 /V  
V 39 /V  
V 40 /V  
V 41 <> "" /V  
V 42 <> "" /V  
V 36 /"END"  
V 43 /V  
V 44 /V  
V 45 /V  
V 46 /V  
V 47 /V  
V 48 <> "" /V  
V 49 <> "" /V  
V 43 /"END"  
V 50 /V

V 51 /V  
V 52 /V  
V 53 /V  
V 54 /V  
V 55 <> "" /V  
V 56 <> "" /V  
V 50 /"END"  
CLOSE

Appendix A

MAILISC.CFG



Mailisc.cfg:

datalist  
nooutput

Appendix A

MAIL.PAS

```
program Mail(input,Out,Out2,output);
```

```
{*-----*}
{* Author: This program was developed by Pamela O. Howard. *}
{* Purpose: This program was built for use in a message *}
{* dissemination system & document routing system. First, *}
{* this program provides a mock mail header for the user. *}
{* Then it accepts the user input and parses it into a *}
{* subject and recipient. This information is placed in *}
{* two files: Hdr.dat and Return.dat. Hdr.dat is used by *}
{* Combine.pas program. EXSYS Professional acquires the *}
{* information that it needs from Return.dat. This info *}
{* is in a special format that EXSYS Professional can *}
{* understand. *}
{*-----*}
```

```
type Str35 = string[35];
    Str15 = string[15];
```

```
var Subject : Str35;
    Recip : Str35;
    Out : text;
    Out2 : text;
    Valu : Str15;
    Slash : boolean;
```

```
procedure Init(var Out : text;var Slash5 : boolean);
{*-----*}
{* This procedure initializes the two text files and the *}
{* boolean value "slash." *}
{*-----*}
```

```
begin
    assign(Out,'Return.dat');
    rewrite(Out);
    assign(Out2,'Hdr.dat');
    rewrite(Out2);
    Slash5 := False;
end;
```

```
procedure GetData(var Sub1 : Str35;var Recip1 : Str35);
{*-----*}
{* The purpose of this procedure is to provide the mock mail *}
{* header to the user and accept user input. *}
{*-----*}
```

```
begin
    writeln('MAIL>SEND');
    write('TO: ');
    readln(Recip1);
    write('SUBJECT: ');
    readln(Sub1);
end;
```

```
procedure SplitData(var Recip4 : Str35;var Valu4 : Str15;
                    var Slash4 : boolean);
{*-----*}
{* The purpose of this procedure is to parse out any info *}
{* from the right side of a slash. For example, if the *}
{* recipient provided by the user is BRANCH CHIEFS/ASPL-A, *}
{*-----*}
```

```

(* this procedure will assign BRANCH CHIEFS to Recip4 and *)
(* ASPL-A to Valu4. *)
(*-----*)

var
  Long : integer;
  Mid : integer;

begin
  if pos('/',Recip4) > 0 then
    begin
      Slash4 := True;
      Mid := pos('/',Recip4);
      Long := length(Recip4);
      Long := Long - Mid;
      Long := Long + 1;
      Valu4 := copy(Recip4,Mid,Long);
      Delete(Valu4,1,1);
      Delete(Recip4,pos('/',Recip4),10);
    end;
  end;

procedure PutData(var Out2 : text; Sub7 : Str35; Recip7 : Str35);
(*-----*)
(* This procedure writes the subject and recipient of the *)
(* message to the text file Hdr.dat. *)
(*-----*)

begin
  writeln(Out2,Sub7);
  writeln(Out2,Recip7);
  close(Out2);
end;

procedure ProcData(var Sub2 : Str35;Recip2 : Str35;Valu2 : Str15;
                   Slash2 : boolean; var Out : text; var Out2 : text);
(*-----*)
(* This procedure translates the subject and recipient *)
(* provided by the user into variable and quantifier values *)
(* that EXSYS Professional can understand. Then it places *)
(* those values into the Return.dat text file. It also *)
(* places one additional value into the Hdr.dat text file. *)
(* This value is used later by the Combine.pas program to *)
(* determine whether time limits need to be added to a list *)
(* of message recipients. *)
(*-----*)

begin
  append(Out2);
  if (Recip2 = 'PRIMARY PROJECT OFFICERS') then
    begin
      writeln(Out,'Q1 ',1);
      writeln(Out2,1);
    end
  else if (Recip2 = 'DIVISION & BRANCH CHIEFS') then
    begin
      writeln(Out,'Q1 ',2);
      writeln(Out,'V12 ',Valu2);
      writeln(Out,'V14 ',Valu2);
      writeln(Out2,2);
    end
  end;
end;

```

```

end
else if (Recip2 = 'DIVISION CHIEFS') then
begin
    writeln(Out,'Q1 ',3);
    writeln(Out,'V12 ',Valu2);

    writeln(Out2,3);
end
else if (Recip2 = 'BRANCH CHIEFS') then
begin
    writeln(Out,'Q1 ',4);
    writeln(Out,'V12 ',Valu2);
    writeln(Out,'V14 ',Valu2);
    writeln(Out2,4);
end
else if (Recip2 = 'DIRECTORATE CHIEFS') then
begin
    writeln(Out,'Q1 ',5);
    if Slash then
        Sub2 := 'PRIVATE';
    writeln(Out2,5);
end
else if (Recip2 = 'PROJECT OFFICERS') then
begin
    writeln(Out,'Q1 ',6);
    writeln(Out,'V5 ',Valu2);
    writeln(Out,'V14 ',Valu2);
    writeln(Out2,6);
end
else if (Recip2 = 'ALL EMPLOYEES') then
begin
    writeln(Out,'Q1 ',7);
    writeln(Out2,7);
end
else if (Recip2 = 'DOC RTNG - DIV') then
begin
    writeln(Out,'Q1 ',8);
    writeln(Out,'V12 ',Valu2);
    writeln(Out2,8);
end
else if (Recip2 = 'DOC RTNG - DIR') then
begin
    writeln(Out,'Q1 ',9);
    writeln(Out,'V12 ',Valu2);
    writeln(Out2,9);
end
else if (Recip2 = 'DOC RTNG - GENERAL') then
begin
    writeln(Out,'Q1 ',10);
    writeln(Out,'V12 ',Valu2);
    writeln(Out2,10);
end;
if (Sub2 = 'FUNDING MEETING') then
    writeln(Out,'Q2 ',1)
else if (Sub2 = 'TRAINING COURSES') then
    writeln(Out,'Q2 ',2)
else if (Sub2 = 'MEETING') then
begin
    writeln(Out,'V14 ',Valu2);
    writeln(Out,'Q2 ',3);

```

```

        end
    else if (Sub2 = 'WEEKLY MEETING') then
        writeln(Out,'Q2 ',4)
    else if (Sub2 = 'PRIVATE') then
        writeln(Out,'Q2 ',5)
    else if (Sub2 = 'SUSPENSE ACTION') then
        writeln(Out,'Q2 ',6)
    else if (Sub2 = 'FYI') then
        writeln(Out,'Q2 ',8)
    else if (Sub2 = 'DOCUMENT APPROVAL') then
        writeln(Out,'Q2 ',9)
    else if (Sub2 = 'WEEKLY ACTIVITY REPORT') then
        writeln(Out,'Q2 ',10)
    else if (Sub2 = 'INFO PAPER') then
        writeln(Out,'Q2 ',11)
    else if (Sub2 = 'MEETING TO ATTEND') then
        writeln(Out,'Q2 ',12)
    else if (Sub2 = 'BLUE BULLET') then
        writeln(Out,'Q2 ',13)
    else
        begin
            Sub2 := 'OTHER';
            writeln(Out,'Q2 ',7);
        end;
    end;

end;

begin (main)
    Init(Out,Slash);
    GetData(Subject,Recip);
    SplitData(Recip,Valu,Slash);
    PutData(Out2,Subject,Recip);
    ProcData(Subject,Recip,Valu,Slash,Out,Out2);
    close(Out);
    close(Out2);
end.

```

Appendix A

COMBINE.PAS

```
program Combine(input, In1, In2, Out, output);
```

```
{*-----*}  
{* Author: This program was developed by Pamela O. Howard.      *}  
{* Purpose: This program was built for use in a message         *}  
{* dissemination system & document routing system. It takes    *}  
{* input from two ASCII files, Hdr.dat and Report.rpt, and      *}  
{* creates a new ASCII file called Newrep.rpt. From Hdr.dat,    *}  
{* it takes information and places this information as the      *}  
{* header of Newrep.rpt. Then it takes information from         *}  
{* Report.rpt and places it into Newrep.rpt as the body of     *}  
{* Newrep.rpt. Based on the qualifier value from Hdr.dat,      *}  
{* time limits are placed on some files.                        *}  
{*-----*}
```

```
type Str35 = string[35];  
    Str15 = string[15];
```

```
var Subject : Str35;  
    Recip : Str35;  
    Out : text;  
    In1 : text;  
    In2 : text;  
    Qualifier : integer;
```

```
procedure Init(var In1 : text; var In2 : text; var Out : text);  
{*-----*}  
{* This procedure initializes the input and output files.      *}  
{*-----*}
```

```
begin  
    assign(In1, 'Hdr.dat');  
    reset(In1);  
    assign(In2, 'Report.rpt');  
    reset(In2);  
    assign(Out, 'NewRep.rpt');  
    rewrite(Out);  
end;
```

```
procedure GetData(var In1 : text; var Sub1 : Str35; var Recip1 : Str35;  
                  var Quall : integer);  
{*-----*}  
{* This procedure acquires input data from the text file      *}  
{* Hdr.dat.                                                    *}  
{*-----*}
```

```
begin  
    readln(In1, Sub1);  
    readln(In1, Recip1);  
    readln(In1, Quall);  
end;
```

```
procedure PutHdr(var Out : text; Sub2 : Str35; Recip2 : Str35);  
{*-----*}  
{* This procedure writes a header to the output text file,    *}  
{* Newrep.rpt. It uses the information obtained above from    *}  
{* Hdr.dat.                                                    *}  
{*-----*}
```

```
begin
```



```

    writeln(Out,'Your message regarding the ',Sub2);
    writeln(Out,'with recipients being ',Recip2);
    writeln(Out,'has generated the following list of EMAIL recipients:');
    writeln(Out);
end;

```

```

procedure PutBody(var Out : text; var In2 : text);
(*-----*)
(* This procedure reads input data from the Report.rpt text *)
(* file and appends this data, unmodified, into the *)
(* Newrep.rpt text file. *)
(*-----*)

```

```

type
    Str50 = string[50];
var
    InpStr : Str50;

begin
    While not eof(In2) Do
        begin
            readln(In2,InpStr);
            writeln(Out,InpStr);
        end;
end;

```

```

procedure PutTimes(var Out : text; var In2 : text);
(*-----*)
(* This procedure reads input data from the Report.rpt text *)
(* file, adds a time limit to the data, and then appends this *)
(* data with the added time limit to the Newrep.rpt text *)
(* file. *)
(*-----*)

```

```

type
    Str50 = string[50];
var
    InpStr : Str50;
    index : integer;

begin
    index := 1;
    While not eof(In2) Do
        begin
            readln(In2,InpStr);
            If (index mod 2 = 1) then
                writeln(Out,InpStr,'1 Day')
            else
                writeln(Out,InpStr);
            index := index + 1;
        end;
end;

```

```

begin {main}
    Init(In1,In2,Out);
    GetData(In1,Subject,Recip,Qualifier);
    PutHdr(Out,Subject,Recip);

```

```

(*-----*)
(* This if/then statement tests the qualifier variable *)

```

```
(* obtained from Hdr.dat.  If the qualifier is 8, 9, or 10, *)
(* then the message is a document routing type message which *)
(* requires time limits to be added. *)
(*-----*)
```

```
  If (Qualifier In [8,9,10]) Then
    PutTimes (Out, In2)
  else
    PutBody (Out, In2);
  close (In1);
  close (In2);
  close (Out);
end.
```

Appendix A

NEWREP.HLP

Newrep.hlp:

The first column contains the EMS user names of people the message should go to.

The second column contains additional information about these people such as their division, project, etc.

Appendix A

QUERY.PRG

Query.prg:

```
*****
*      Program to query database of usernames      *
*      for Mail Dissemination System and          *
*      Document Routing System, Masters Project    *
*
*      Date: Spring 1990                          *
*      Developed by: Pamela O. Howard, Milam Aiken, *
*                  Chih-Ping Wei                  *
*****

* Clear screen, turn off options
CLEAR
@ 10,35 SAY "Working"
SET ESCAPE OFF
* SET DEFAULT TO A:
SET BELL OFF
SET CONSOLE OFF
SET SAFETY OFF
SET TALK OFF
SET ECHO OFF
* erase final report if it exists
IF FILE('REPORT.RPT')
  ERASE REPORT.RPT
ENDIF
* Clean out temporary databases
SELECT 3
  USE TEMP
  ZAP
  SELECT 4
  USE REPORT
  ZAP
SELECT 1
USE QUERY
  ZAP
* Input query from expert system and process it
APPEND FROM PASS.DAT SDF
GO TOP
DO WHILE .NOT. EOF()
STORE CONDITION TO TN1
SKIP
STORE CONDITION TO RF1
SKIP
STORE CONDITION TO RF2
SKIP
IF CONDITION <> 'END'
  STORE CONDITION TO MF1
  SKIP
  STORE CONDITION TO MV1
  SKIP
  IF CONDITION <> 'END'
    STORE CONDITION TO MF2
    SKIP
    STORE CONDITION TO MV2
    SKIP
  SELECT 2
  USE &TN1
  COPY TO TEMP.TXT FIELDS &RF1,&RF2 FOR &MF1=MV1 .AND. &MF2=MV2 SDF
ELSE
```

```

        SELECT 2
        USE &TN1
        COPY TO TEMP.TXT FIELDS &RF1,&RF2 FOR &MF1=MV1 SDF
    ENDIF
    ELSE
        SELECT 2
        USE &TN1
        COPY TO TEMP.TXT FIELDS &RF1,&RF2 SDF
    ENDIF

*   Append output from TEMP.TXT to REPORT.RPT
*   ---- Section changed to eliminate DOS calls ----

*       Old code

*       First attempt   --- works but puts copied file names on screen

* run echo off
* if file('report.txt')
*     run copy report.txt + temp.txt
*     erase temp.txt
* else
*     run rename temp.txt report.txt
* endif

*       Second attempt   --- does not append correctly

* if file('report.rpt')
*     select 3
*     use temp
*     append from temp.txt sdf
*     copy to report.rpt sdf
*     erase temp.txt
* else
*     rename temp.txt report.rpt
* endif

*       Last attempt

*       Put temp.txt in temp.dbf file
*       and append temp.dbf info to report.dbf

        SELECT 3
        USE TEMP
        APPEND FROM TEMP.TXT SDF
        GO TOP
        DO WHILE .NOT. EOF()
            STORE USERNAME TO PART1
            STORE INFO TO PART2
            SELECT REPORT
            APPEND BLANK
            REPLACE USER WITH PART1
            REPLACE OTHER WITH PART2
            SELECT TEMP
            SKIP
        ENDDO

*   Erase temporary files
SELECT TEMP
ZAP
ERASE TEMP.TXT

```

SELECT 1

SKIP

ENDDO

\* Place summary in REPORT.RPT

SELECT 4

USE REPORT

COPY TO REPORT.RPT SDF

CLOSE ALL

QUIT



# Appendix A

READ.ME

Copy the following files from the disks into the same directory as the runtime version of EXSYS Professional (version 1.1.4). At the DOS prompt, type ``GO.`` Then, follow the instructions in the User's guide.

These files must be present to run the Message Dissemination System (MDS) and the Document Routing System (DRS).

GO.BAT            A batch file that calls the necessary sequence of programs.

REPORT.EXE        A report formatting program developed by Mike Morrison.  
Syntax: REPORT filename.RPT "Header Info"  
See batch file for use here.

NEWREP.HLP        Note: REPORT.EXE must have a file called filename.HLP in order to work. This is a popup help window. This .hlp file is included with the program for that purpose.

TEMP.DBF          These files must be present for the query program to work.  
REPORT.DBF  
QUERY.DBF

MAILISC.TXT       These files are expert system files that work in conjunction with EXSYS Professional. They were built specifically to support USAISC.  
MAILISC.RUL  
MAILISC.CFG  
MAILISC.CMD  
MAILISC.OUT

MAIL.PAS          This is the source code and the executable code for the Pascal programs that are called to work with the MDS and DRS. They are called by the batch file.  
MAIL.EXE  
COMBINE.PAS  
COMBINE.EXE

QUERY.PRG         This is the source code and the executable code for the dBase III plus program which builds queries and provides a list of EMAIL recipients.  
QUERY.EXE

EXSYSP.OVL        These are the EXSYS Professional (version 1.1.4) files required for the program to operate. Since EXSYS Professional is a copy protected program, any future users must purchase a copy of the program. It cannot be included with this research project.  
EXSYSP.HLP  
EXSYSP.COM  
EXSYSP.EXE

Return.dat        Do not be alarmed to find these files after running the program. They are built by the various executable programs for transferring or displaying data.  
Report.rpt  
Report.txt  
Newrep.rpt  
Hdr.dat  
Pass.dat

ISCTOP.DBF        These are the .dbf files for the dBase III plus database. They were created exclusively for use with this program.  
ISCDIR.DBF  
ISCDIV.DBF  
ISCBR.DBF  
ISCEMP.DBF  
ISCPROJ.DBF  
ISCBRPRO.DBF  
ISCEMPPR.DBF

DIREMP.DBF  
DVEMP.DBF  
BREMP.DBF  
BRPROCH.DBF  
BDD.DBF

These .dbf files were created through joining  
one or more of the above files.

## **9 Appendix B - Figures**

1. Figure 1: KMS/MDS/DRS Architecture
2. Figure 2: A Database/Knowledge base Coupled System
3. Figure 3: SOM Chart for USAISC

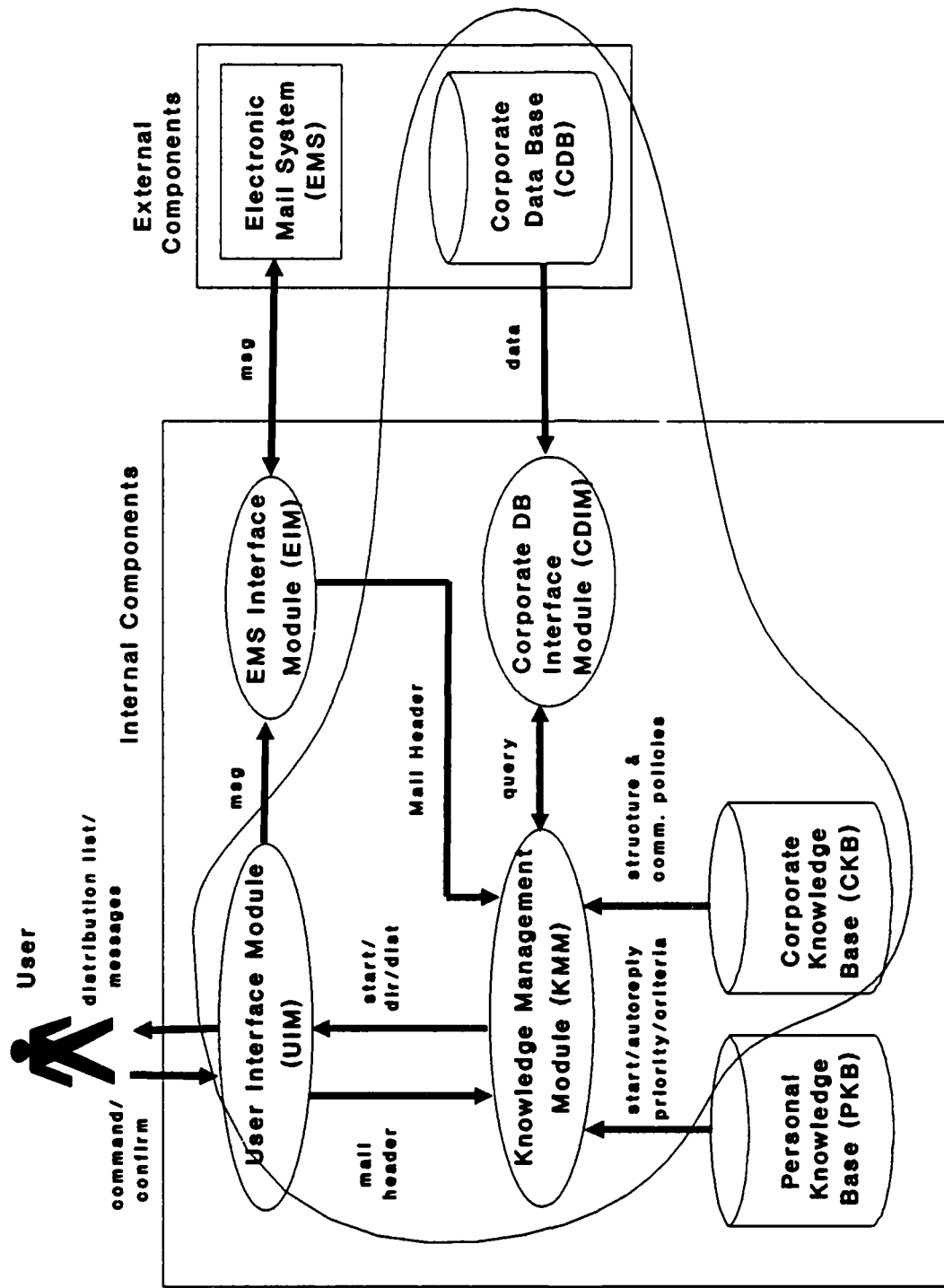
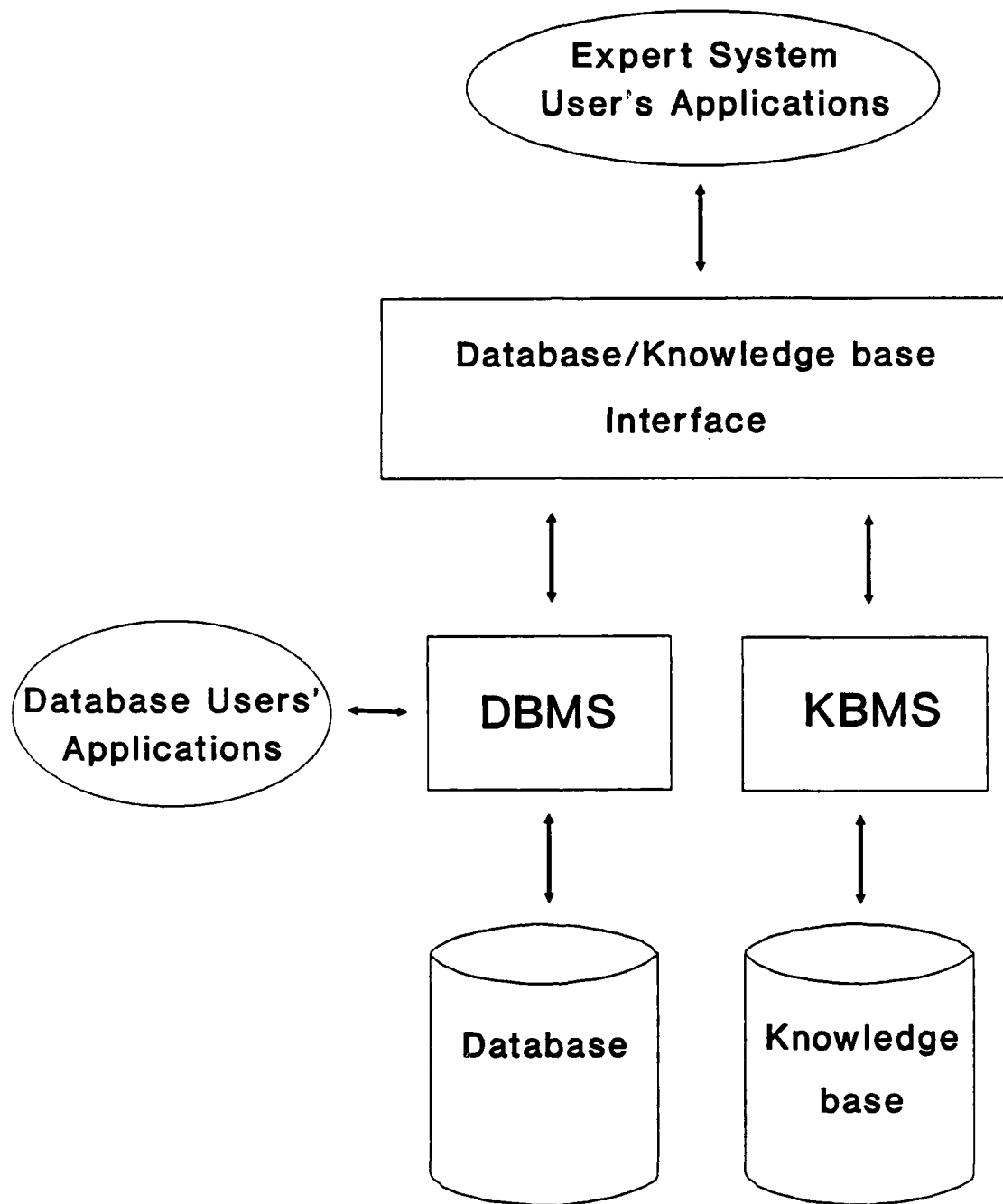


Figure 1: KMS/MDS/DRS Architecture [7]



**Figure 2: A Database/Knowledge Base Coupled System**



## 10 Appendix C - User's Guide

### 10.1 Introduction

The Mail Dissemination System (MDS) and the Document Routing System (DRS) are automated tools that aid action officers in selecting the proper recipients for a given EMAIL message. They are most useful for those situations when the user wants to send a message to several people in the organization.

These systems were designed to work in an embedded fashion, behind an EMS system. So for example, if the EMS system cannot understand the mail header responses, the TO and SUBJECT lines, the EMS would call the batch file which would execute the various files involved with the MDS and DRS. After that, the user will see a "working" message. Once the programs have finished executing, the proposed list of message recipients is placed on the computer screen.

### 10.2 System Requirements

System requirements for correct operation of this software follow:

- Operating System: DOS 2.0 or higher.
- CPU: 80286 (an AT class personal computer).
- Speed: 10 MHZ or faster.
- RAM: 640 kilobytes.
- Monitor: Color monitor is recommended, but not required.
- Printer: Not required for system operation.
- Commercial software: A runtime version of EXSYS Professional, version 1.1.4, is required for this software to operate correctly.

### 10.3 Operation

Copy all of the files from the two disks in the back of this report into the same directory as EXSYS Professional (version 1.1.4, at least). At the DOS prompt, type go. The go.bat file will execute, and "TO:" will soon appear on the screen. Type in *exactly* one of the "TO:" responses listed below. Then, a "SUBJECT:" prompt will appear. Type in *exactly* the associated "SUBJECT:" response listed below. Shortly thereafter, a



"working" message will appear on the screen. Following that, the screen containing the recipient computer id names, etc. will appear.

Due to the limited number of situations which these systems can handle as well as to a lack of error correction facilities, only a given set of TO and SUBJECT responses will result in correct operation of the system. The following list represents the subset of responses which these systems are currently designed to understand.

1. TO: PRIMARY PROJECT OFFICERS  
SUBJECT: FUNDING MEETING
2. TO: DIVISION & BRANCH CHIEFS/ASPL (any directorate designation will work)  
SUBJECT: TRAINING COURSES
3. TO: DIVISION CHIEFS/ASPL (any directorate designation will work)  
SUBJECT: MEETING
4. TO: DIVISION CHIEFS/ASRM (any directorate designation will work)  
SUBJECT: TSQ-XXX (any subject will work)
5. TO: BRANCH CHIEFS/ASPL-A (any division designation will work)  
SUBJECT: WEEKLY MEETING
6. TO: BRANCH CHIEFS/ASRM-PB (any division designation will work)  
SUBJECT: SUSPENSE ACTION
7. TO: DIRECTORATE CHIEFS/PRIV (indicates privacy)  
SUBJECT: RIF (any subject will work)
8. TO: DIRECTORATE CHIEFS  
SUBJECT: ANYSUB (any subject will work)
9. TO: PROJECT OFFICERS/AWIS (any valid project id will work)  
SUBJECT: MEETING (any subject will work)
10. TO: ALL EMPLOYEES  
SUBJECT: SECURITY BRIEFING (any subject will work)
11. TO: DOC RTNG - DIV/ASPL-AA (any ASPL branch designation will work)  
SUBJECT: FYI
12. TO: DOC RTNG - DIR/ASPL-AA (any ASPL branch designation will work)  
SUBJECT: DOCUMENT APPROVAL
13. TO: DOC RTNG - DIR/ASPL-AA (any ASPL branch designation will work)  
SUBJECT: WEEKLY ACTIVITY REPORT

14. TO: DOC RTNG - GENERAL/ASPL-AA (any ASPL branch designation will work)

SUBJECT: INFO PAPER

15. TO: DOC RTNG - GENERAL/ASPL-AA (any ASPL branch designation will work)

SUBJECT: BLUE BULLET

## 10.4 Sample Runs

### 10.4.1 Sample MDS Run

First Screen (user types "go"):

```
C:\AIMAIL\EXSYSP>GO
```

Second Screen (user types in recipient):

```
MAIL>SEND  
TO: DIVISION & BRANCH CHIEFS/ASPL
```

Third Screen (user types in subject):

```
MAIL>SEND  
TO: DIVISION & BRANCH CHIEFS/ASPL  
SUBJECT: TRAINING COURSES
```

Fourth Screen:

Working

Fifth Screen (recipient listing):

Message Dissemination System

Scrollable Window - Press Esc to exit

Your message regarding the TRAINING COURSES  
with recipients being DIVISION & BRANCH CHIEFS  
has generated the following list of EMAIL recipients:

|               |       |
|---------------|-------|
| HSLATER       | CHIEF |
| BCROUCH       | CHIEF |
| ZBROWN        | CHIEF |
| JMOFFETT      | CHIEF |
| HWILSON       | CHIEF |
| JDUGAS        | CHIEF |
| MBROWN        | CHIEF |
| DHIGGINS      | CHIEF |
| GPLEGGENKUHLE | CHIEF |
| RFURLONG      | CHIEF |

Esc or F10 - Continue

F1 - HELP

#### 10.4.2 Sample DRS Run

First Screen (user types "go"):

```
C:\AIMAIL\EXSYSP>GO
```

Second Screen (user types in recipient):

```
MAIL>SEND  
TO: DOC RTNG - GENERAL/ASPL-AA
```

Third Screen (user types in subject):

```
MAIL>SEND  
TO: DOC RTNG - GENERAL/ASPL-AA  
SUBJECT: BLUE BULLET
```

Fourth Screen:

Working

Fifth Screen (recipient listing):

Message Dissemination System

Scrollable Window - Press Esc to exit

Your message regarding the BLUE BULLET  
with recipients being DOC RTNG - GENERAL  
has generated the following list of EMAIL recipients:

|           |       |       |
|-----------|-------|-------|
| JMOFFETT  | CHIEF | 1 Day |
| RSUPKUPA  | SECY  |       |
| HSLATER   | CHIEF | 1 Day |
| JNESBITT  | SECY  |       |
| KTHOMAS   | CHIEF | 1 Day |
| PBAINES   | SECY  |       |
| TROGERS   | CHIEF | 1 Day |
| JDENWANGO | SECY  |       |

Esc or F10 - Continue

F1 - HELP

## References

- [1] Higa, Kunihiko and Olivia R. Liu Sheng. "An Object-Oriented Methodology for Database/Knowledgebase Coupling: An Implementation of the Structured Entity Model in Nexpert System," ACM SIGBDP, May, 1989.
- [2] Higa, Kunihiko and Olivia R. Liu Sheng. "An Object-Oriented Methodology for End-user Logical Database Design: The Structured Entity Model Approach," Department of Management Information Systems, College of Business and Public Administration, University of Arizona, Tucson, AZ.
- [3] Higa, Kunihiko, Milam W. Aiken, and Olivia R. Liu Sheng. "A Knowledge-Based Message Dissemination System: Design and Implementation," Department of Management Information Systems, College of Business and Public Administration, University of Arizona, Tucson, AZ.
- [4] Higa, Kunihiko, Milam W. Aiken, and Olivia R. Liu Sheng. "A Knowledge-Based Mail System for Collaborative Work," Department of Management Information Systems, College of Business and Public Administration, University of Arizona, Tucson, AZ.
- [5] Howard, Pamela O. "Requirements Analysis for a KMS for an Army Office," working paper, Department of Management Information Systems, College of Business and Public Administration, University of Arizona, Tucson, AZ.
- [6] Howard, Pamela O., Richard Orwig, Chih-Ping Wei, and Hsiao-Yu Wu. "Prototype Message Dissemination System (MDS) for the MIS Department," class paper, MIS 580, December 11, 1989, Department of Management Information Systems, College of Business and Public Administration, University of Arizona, Tucson, AZ.
- [7] Motiwalla, Luvai F., Kunihiko Higa, Olivia R. Liu Sheng, and Jay F. Nunamaker, Jr. "A Knowledge-Based Mail System to Support Management Activities," Department of Management Information Systems, College of Business and Public Administration, University of Arizona, Tucson, AZ.
- [8] Motiwalla, Luvai. "A Knowledge-Based Electronic Messaging System: Framework, Design, Prototype Development, and Validation," Unpublished Ph.D. Dissertation, Department of Management Information Systems, College of Business and Public Administration, University of Arizona, Tucson, AZ.